PLOS ONE

RESEARCH ARTICLE

A Hidden Markov Model for Urban-Scale Traffic Estimation Using Floating Car Data

Xiaomeng Wang[1,2], Ling Peng[1]*, Tianhe Chi[1], Mengzhu Li[3], Xiaojing Yao[1,2], Jing Shao[1,2]

1 Institute of Remote Sensing and Digital Earth, Chinese Academy of Sciences, Beijing, China, **2** University of Chinese Academy of Sciences, Chinese Academy of Sciences, Beijing, China, **3** College of Economics and Management, Southwest University, Chongqing, China

* pengl20158@163.com

Citation: Wang X, Peng L, Chi T, Li M, Yao X, Shao J (2015) A Hidden Markov Model for Urban-Scale Traffic Estimation Using Floating Car Data. PLoS ONE 10(12): e0145348. doi:10.1371/journal.pone.0145348

Editor: Tieqiao Tang, Beihang University, CHINA

Received: August 25, 2015

Accepted: December 2, 2015

Published: December 28, 2015

Data Availability Statement: Data have been deposited to Figshare: http://dx.doi.org/10.6084/m9.figshare.1618856.

Funding: TC is supported by a National Key Technology Support Program (2015BAJ02B00) and Ministry of Science and Technology Policy Guidance Project (2011FU125Z24). URL: http://program.most.gov.cn/. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing Interests: The authors have declared that no competing interests exist.

Abstract

Urban-scale traffic monitoring plays a vital role in reducing traffic congestion. Owing to its low cost and wide coverage, floating car data (FCD) serves as a novel approach to collecting traffic data. However, sparse probe data represents the vast majority of the data available on arterial roads in most urban environments. In order to overcome the problem of data sparseness, this paper proposes a hidden Markov model (HMM)-based traffic estimation model, in which the traffic condition on a road segment is considered as a hidden state that can be estimated according to the conditions of road segments having similar traffic characteristics. An algorithm based on clustering and pattern mining rather than on adjacency relationships is proposed to find clusters with road segments having similar traffic characteristics. A multi-clustering strategy is adopted to achieve a trade-off between clustering accuracy and coverage. Finally, the proposed model is designed and implemented on the basis of a real-time algorithm. Results of experiments based on real FCD confirm the applicability, accuracy, and efficiency of the model. In addition, the results indicate that the model is practicable for traffic estimation on urban arterials and works well even when more than 70% of the probe data are missing.

Introduction

Traffic congestion has become a severe problem in metropolises, resulting in widespread wastage of time and energy [1]. Traffic monitoring and estimation is an important method for obtaining information on traffic conditions; thus, it plays a vital role in reducing traffic congestion [2]. Static sensors (inductive loop detectors [3], video cameras [4], etc.), deployed at fixed locations on roads, are used to detect traffic state (e.g., flow velocity and traffic density). However, it is difficult for these traditional approaches to cover all roads because they involve extensive infrastructure deployment and high maintenance costs [5].

With the rapid development of mobile technologies, recent years have witnessed the emergence of a new method known as floating car data (FCD) for collecting valuable real-time information on traffic conditions. In this method, vehicles (e.g. taxis and buses) equipped with global positioning systems (GPS), accelerometers, and other sensors can provide data such as position, velocity, and acceleration of the vehicle. FCD is not as expensive as traditional data

acquisition methods because it requires no dedicated infrastructure. Moreover, it has the potential to provide good spatiotemporal coverage of the transportation network and useful data given a certain penetration rate in the population [6, 7].

In some previous studies, FCD has been used to estimate traffic conditions on highways, providing good results with a low penetration rate (1%–3%) [7–9]. Compared to traffic estimation on freeways, traffic estimation on arterials is more complex because of the traffic lights and intersections, and it requires a greater number of samples for analysis. Several researches have discussed the minimum penetration rate required. For example, Breitenberger et al. [10] proposed a penetration rate of 10% on arterial and urban roads. In addition, Vandenberghe et al. [11] discussed the maximum sample interval and the maximum transmission interval of aggregated samples, where the former defines the time between two consecutive FCD samples captured by the same floating car and the latter defines the time between two consecutive server uploads of all new samples by a floating car.

However, it is difficult for FCD to meet the sampling requirements in practice, and the distribution of observed probe data may be sparse and uneven. Traffic state estimation using sparse probe data has not been explored extensively. Herring et al. [12] proposed a probabilistic modeling framework for estimating arterial travel time distribution using sparse probe data. They modeled the evolution of traffic states as a coupled hidden Markov model (HMM), in which the traffic states of nearby road segments are correlated and evolve over time in a Markov manner. The present study differs from their study in that it considers links with similar traffic conditions instead of adjacent links of the road network, which may improve the modeling accuracy. Yanmin et al. [13] revealed the hidden structures within the traffic conditions of a road network using principal component analysis (PCA) and proposed a compressive sensing-based algorithm for obtaining the missing traffic conditions. However, they simply developed an offline data analytics algorithm that cannot be applied to real-time traffic estimation.

The present study proposes an HMM-based model that focuses on overcoming the problem of data sparseness for traffic estimation using FCD. It is assumed that the traffic state of a road segment is invisible and that each road segment belongs to a cluster of road segments having similar traffic characteristics. The traffic conditions of the other road segments in the cluster are considered as observations, based on which an HMM can be constructed. An algorithm based on clustering and pattern mining is proposed to find all road segment clusters in which segments have similar traffic characteristics, and a multi-clustering strategy is adopted to achieve a trade-off between clustering accuracy and coverage. Through data analysis, two exponential distribution functions are used for computing emission probability and transition probability. Finally, a real-time estimation algorithm is developed for online traffic application. The results of extensive experiments conducted using real floating car data show that our model works well even when more than 70% of the probe data are missing.

The remainder of this paper is organized as follows. Section 2 describes the problem of traffic estimation using sparse probe data. Section 3 discusses the construction of an HMM-based traffic estimation model, outlines the main steps of the proposed approach, and presents an algorithm for real-time traffic estimation. Section 4 describes the implementation of the proposed model and as well as a case study for assessing the accuracy of the model. Finally, Section 5 summarizes our findings and concludes the paper.

Problem Description

There are numerous floating cars running on the roads. They upload their state information, such as location, speed, and direction, from time to time. The state of a floating car at time t is expressed as $s<id, l, v, t>$, where id, l, and v denote the ID, location, and speed, respectively, of

the vehicle. A road network G is divided into a set of road segments, $R = \{r_n \,|n = 1, 2, \ldots, N\}$, by intersections. A map-matching algorithm is used to find the road segment on which the vehicle is traveling at time t. In order to facilitate statistical analysis, a set of predefined time slots, $T = \{t_m \,|m = 1, 2, \ldots, M\}$, instead of continuous time, is employed for traffic condition estimation. Then, the state of vehicle s is converted into $s<id, r, v, t>$.

In the field of traffic engineering, several metrics have been proposed for quantifying the traffic condition of a link, such as speed [14], density [15], flow [16], and queues at intersections [17]. Furthermore, Many traffic flow models [18–29] have been proposed to study complex traffic conditions. The present study employs the velocity of the traffic flow on a road segment, as in some previous studies [1, 13, 14, 30, 31]. The floating cars are part of the traffic flow; hence, it is reasonable to consider their speed as the speed of the traffic flow. The speed of the traffic flow on segment n at time slot m, $x^{n,m}$, is approximated as the average speed of all vehicles moving within the traffic flow on this road segment at time slot m. Then, the traffic condition of the road network can be expressed by matrix X as follows:

$$X = \begin{bmatrix} x^{1,1} & \cdots & x^{1,M} \\ \vdots & x^{r,t} & \vdots \\ x^{N,1} & \cdots & x^{N,M} \end{bmatrix} \tag{1}$$

Here, the row $X_r = \{x^{r,m} \,|m = 1, 2, \ldots, M\}$ represents the traffic condition sequence of road segment r over time. Because of the randomness and unevenness of the floating car data, it is difficult to obtain a complete traffic condition matrix, as there are many spatiotemporal vacancies with no probe measurements. As shown in Fig 1, for the traffic condition sequence of a road segment, there may be some sub-sequences without sample data. Hence, in this study, the main objective of traffic estimation is to estimate the values of these missing states, which can approximate the true states.

Methods

Estimation model based on HMM

In this paper, an HMM-based estimation model is proposed to estimate missing traffic state sequences. An HHM, which is based on the concept of Markov process and Markov chain, is characterized by five elements: observation, hidden state, state transition probability, emission probability, and initial state.

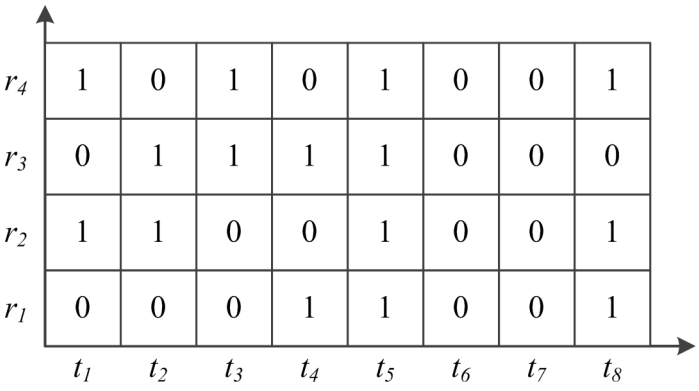


Fig 1. Traffic condition sequences. The value 1 indicates the presence of sample data and 0 indicates the absence of sample data at a time slot.

doi:10.1371/journal.pone.0145348.g001

The first step in HMM construction is to establish the observations and hidden states of the model. In this study, the traffic condition of the target road segment r at time slot t is the hidden state $x^{r,t}$. It is assumed that the road segment r belongs to a cluster C in which all road segments have similar traffic characteristics. Then, the observations $y^{r,t}$ are defined as the traffic conditions of the other road segments in the cluster. In some previous studies [16, 32], adjacent road segments have been assumed to be correlated with each other. However, in practice, such an assumption may be not very accurate. A method based on clustering and frequent pattern mining is proposed in Section 3.2 in order to find clusters having road segments with similar traffic characteristics. In this study, speed, which is a continuous variable, is employed as the traffic condition; thus, the hidden state has an infinite number of values. Therefore, it is necessary to select finite candidate states for the HMM process. The state value range at t can be limited according to the observation $y^{r,t}$ and previous state $x^{r,t-1}$; then, the range should be discretized to a candidate state set $CS^{r,t} = \{x_i^{r,t} | i = 1, 2, \ldots, k\}$.

The emission probability, $Pr(y^{r,t}|x^{r,t})$, is the likelihood of observing the traffic condition $y^{r,t}$ conditional on the traffic condition $x^{r,t}$ being the true condition of the road segment r at time slot t. The transition probability, $Pr(x^{r,t}, x^{r,t+1})$, is the probability that the traffic condition of the road segment r will transform from a state $x^{r,t}$ at time t to another state $x^{r,t+1}$ at time $t+1$. The methods for measuring and calculating the emission probability and transition probability are discussed in Section 3.3.

The HMM sequentially generates candidate traffic condition sequences and evaluates them on the basis of their likelihood, which is measured by the joint probability (Fig 2). Past hypotheses of the solution are extended to account for new observations over time. Then, the surviving sequence with the highest joint probability is selected from among the remaining candidates of the previous stage as the final solution. The joint probability is expressed as

$$J^{r,t+1} = Pr(y^{r,t+1}|x^{r,t+1}) \max_{x^{r,t} \in CS^{r,t}} \{Pr(x^{r,t}, x^{r,t+1}) J^{r,t}\} \quad (2)$$

where $J^{r,1} = Pr(y^{r,1}|x^{r,1})$ and $CS^{r,t}$ denotes the set of candidate states of the road segment r at time slot t. After the HMM process, the last traffic condition sequence with the maximum joint probability can be found: $x^{r,T} = argmax_{x^{r,T} \in CS^{r,T}} \{J_T\}$. Then, the system works backwards to find the traffic condition sequence $x_{T-1}, \ldots, x_1$ of the road segment.

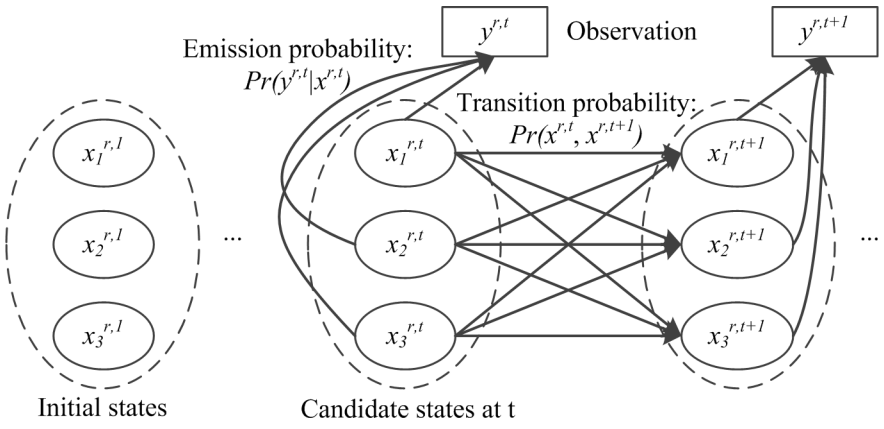


Fig 2. HMM process of traffic condition estimation.

doi:10.1371/journal.pone.0145348.g002

Traffic similarity analysis

Clustering analysis. In this study, the traffic condition sequence, $X_r = \{x^{r,t} | t = 1, 2, \ldots, M\}$, is considered as the traffic characteristic of the road segment r over a period of time. A spectral clustering algorithm is adopted to divide the road segment set into clusters based on historical data, and the road segments in the same cluster have similar traffic characteristics.

In the spectral clustering algorithm, the set of points in an arbitrary feature space can be represented as a complete weighted undirected graph $G(V, E)$. The vertices of the graph G are the points in the feature space and the weight w_{ij} of an edge (v_i, v_j) in E is a measure of the similarity between vertex v_i and v_j. In this context, we can formulate the clustering problem as a graph-partitioning problem that requires partitions $V_1, V_2 \ldots, V_k$ of the vertex set V according to some measure; then, the vertices in any set V_i have a high degree of similarity, and the vertices in two different sets V_i, V_j have a low degree of similarity.

For road segment clustering, the road segments are considered as vertices of the graph G. The weight w_{ij} between the road segments i and j is the difference between the traffic characteristics of the two road segments; it can be expressed by a Euclidean distance as follows:

$$w_{ij} = \sqrt{\sum_{t=1}^{M} (x^{i,t} - x^{j,t})^2} \tag{3}$$

where M is the number of time slots in the traffic condition sequence and $x^{i,t}$ is the traffic condition of road segment i at time slot t. A normalized spectral clustering algorithm (Box 1) is constructed according to previous research [33]:

In practice, it is difficult to determine a suitable number of clusters for road segment clustering. Therefore, a modified clustering algorithm is proposed, and the average weight w_{av} of a cluster, instead of the cluster number k, is set as a constraint for controlling the clustering process. In order to simplify the computation, w_{av} is defined as the average weight between the centroid of a cluster and other objects. The centroid is given by

$$vc_k = \frac{1}{|V_k|} \sum_{v_i \in V_k} v_i \tag{4}$$

where V_k denotes the vertices of the k-th cluster, $|V_k|$ is the number of vertices in V_k, v_i is the i-th vertex of V_k, and vc_k is the centroid of V_k,. Then, the average weight of V_k can be expressed

Box 1. Spectral clustering algorithm.

Algorithm 1 SpectralClustering: spectral clustering
Input: G(V, E): Traffic condition graph; k: Number of clusters;
Output: $C = \{V_1, V_2, \ldots V_k\}$: clusters;
1: Get weighted adjacency matrix W of $G(V, E)$;
2: Calculate degree matrix D of W;
3: $L \leftarrow I - D^{-\frac{1}{2}} \cdot W \cdot D^{-\frac{1}{2}}$; //Compute normalized Laplacian
3: Compute the first k eigenvectors $u_1, \ldots u_k$ of L;
4: Let U be the matrix containing the vectors $u_1, \ldots, u_k$ as columns;
5: Use the k-means algorithm to cluster U, then get the clusters C;
6: return C;

as

$$w_{av} = \frac{1}{|V_k|} \sum_{v_i \in V_k} \|v_i - vc_k\| \tag{5}$$

In the algorithm (Box 2) based on the constraint ω, the vertices of G are divided into small clusters step by step, and the cluster whose w_{av} is greater than the threshold value ω should be divided into smaller clusters until the constraint is met. Because clusters with only one object are meaningless, it is reasonable to set a minimum number of objects in a cluster (N_{min}). A small value k is set as the number of clusters in every clustering step. In this study, both k and N_{min} are set as 2.

Pattern mining. To avoid coincidental clusters, it is reasonable to perform clustering multiple times for different days and find frequent clusters with a better representation of traffic similarity between road segments. In general, the traffic condition exhibits different patterns on weekdays and weekends. As shown in Fig 3, the traffic conditions of arterial roads in Beijing have similar characteristics on weekdays but significantly different characteristics on weekends. Therefore, the traffic conditions should be discussed separately.

The road segment set, $R = \{r_n \,|n = 1, 2, \ldots, N\}$, is divided into cluster set $C_d = \{R_k \,|k = 1, 2, \ldots, K\}$ according to the traffic condition of the d-th day using the clustering algorithm proposed in Section 3.2.1. The cluster set list, $L = \{C_d \,|d = 1, 2, \ldots, D\}$, contains all clusters of the last D days (weekdays or weekends) from the target day. The objective of the frequent pattern mining approach adopted in this study is to find the frequent cluster set, $P = \{R_j \subset R \,|j = 1, 2, \ldots, J\}$, where the cluster R_j appears frequently in L. An indicator function is used to indicate whether R_j appears in the cluster set C_d:

$$f(R_j, C_d) = \begin{cases} 1 \text{ if } \exists R_k \in C_d, \text{ let } R_j \subset R_k \\ 0 \text{ if } \nexists R_k \in C_d, \text{ let } R_j \subset R_k \end{cases} \tag{6}$$

Box 2. Clustering algorithm based on constraint.

Algorithm 2 ConstraintClustering: Clustering based on constraint
Input: $G(V, E)$: Traffic condition graph; ω: Threshold value; k: Number of clusters at every step; N_{min}: Minimum number of objects in a cluster;
Output: $C = \{V_1, V_2, \ldots V_K\}$: *clusters;*

```
if |V|>N_min and |V|>k
  C ← SpectralClustering(G, k);
  C_temp ← ∅;
  for i ← 1 to k do
    if the average weight w_av of V_i is greater than ω
      Get sub-graph G_i from G corresponding to V_i;
      C_temp ← C_temp ∪ SpectralClustering(G_i, k);
    else
      C_temp ← C_temp ∪ {V_i};
    end if
  end for
  C ← C_temp;
else
  C ← {V};
end if
return C;
```

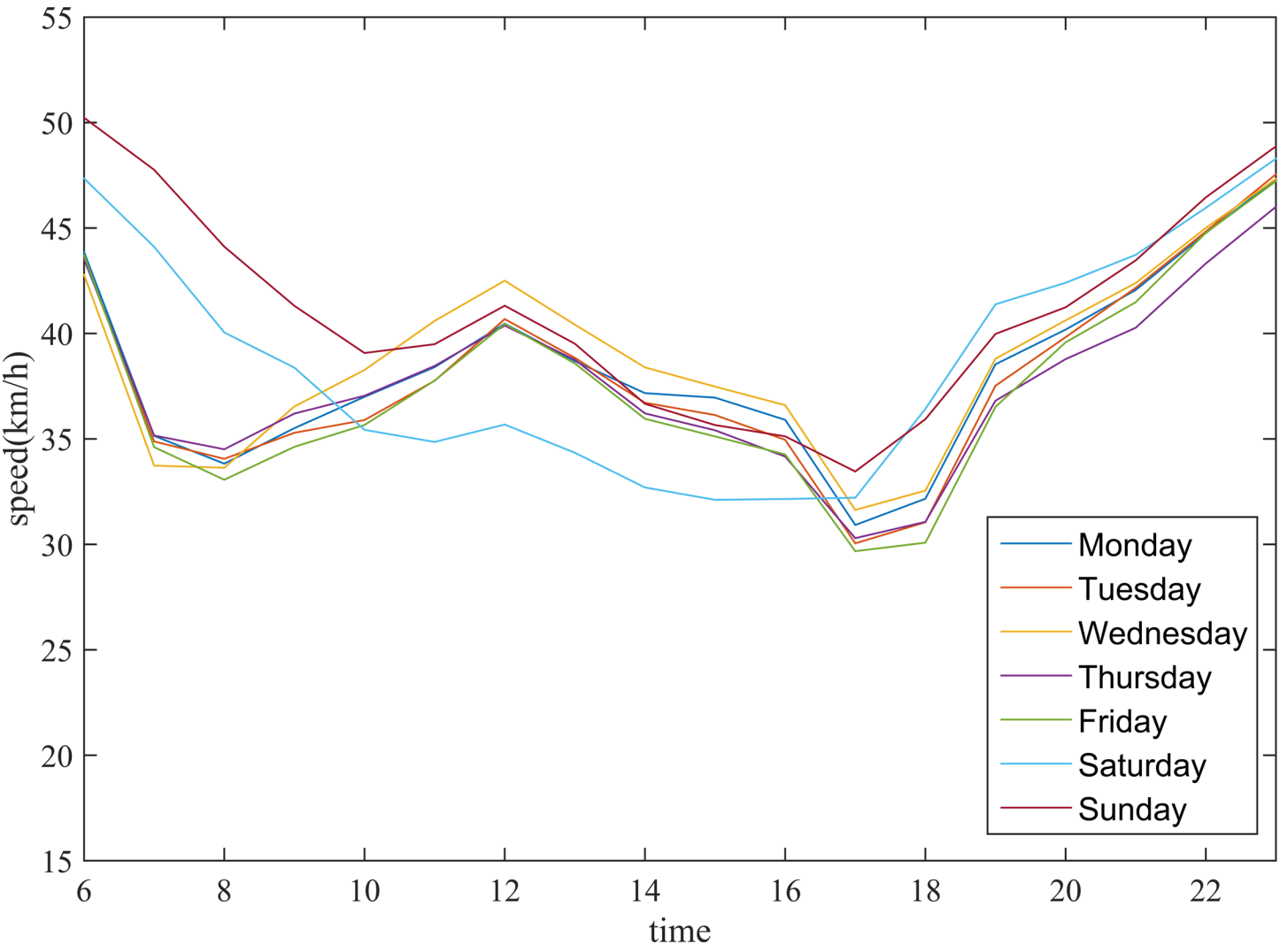


Fig 3. Average speed of arterial roads in Beijing.

doi:10.1371/journal.pone.0145348.g003

In this study, the number of times the cluster R_j appears in the cluster set L is defined as the support, and it can be calculated as follows:

$$support(R_j, L) = \sum_{C_d \in L} f(R_j, C_d) \tag{7}$$

The frequent cluster must meet the minimum support, Sup_{min}; then, the frequent cluster set can be defined as follows:

$$P(L) = \{R_j \subset R | j = 1, 2, \ldots J, and\ support(R_j, L) \geq Sup_{min}\} \tag{8}$$

It is difficult for traditional pattern mining algorithms (e.g., the Apriori algorithm) to compute and find the frequent clusters, as the number of road segments and clusters is extremely large. To overcome this problem, a frequent pattern mining approach based on intersection is proposed. The intersection between two cluster sets is expressed as

$$intersection(C_1, C_2) = \{R_k | R_k = R_i \cap R_j, R_i \subset C_1, R_j \subset C_2, and\ |R_k| > 1\} \tag{9}$$

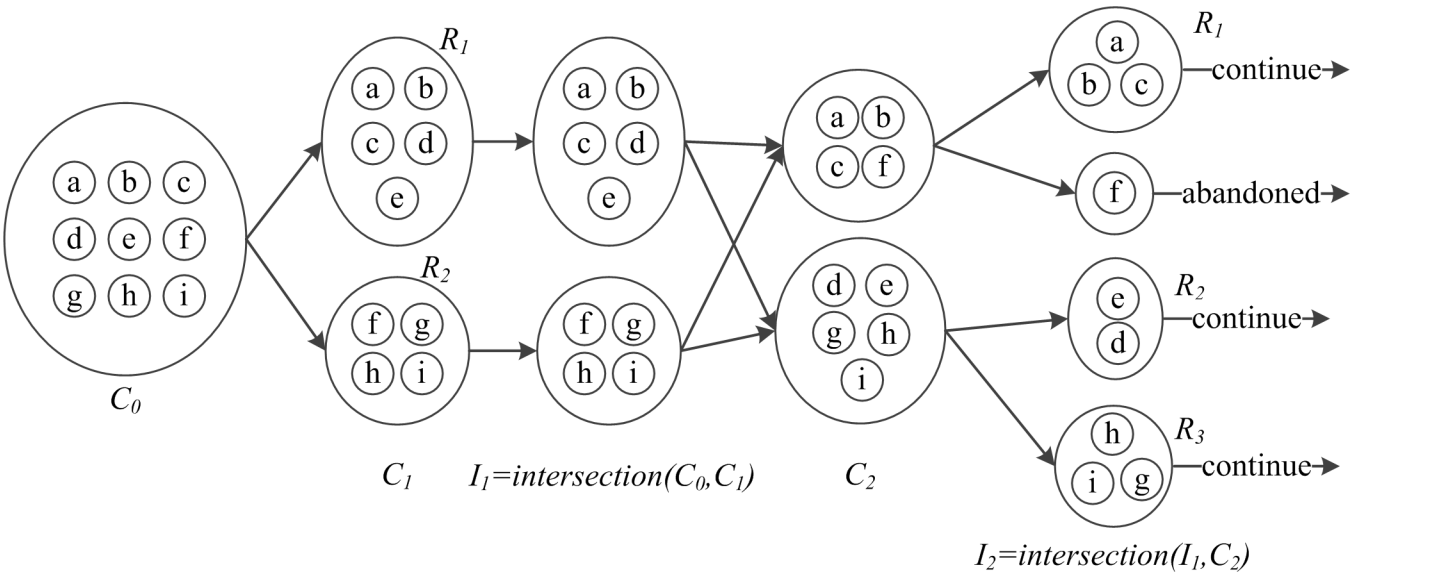


Fig 4. Frequent clusters are obtained by gradual intersection.

doi:10.1371/journal.pone.0145348.g004

where C_1 and C_2 are the cluster sets of two days, R_i and R_j are arbitrary clusters of sets C_1 and C_2, respectively, and R_k is the intersection of R_i and R_j. Note that R_k is meaningful for estimation only if it includes more than one road segment. Therefore, in the intersection process, the cluster R_k will be discarded when $|R_k|<2$. As shown in Fig 4, the frequent cluster set can be obtained by gradual intersection.

For a cluster set list L_i, which has i cluster sets, the frequent clusters that appear i times in L_i can be found by a recursive algorithm based on intersection; this algorithm can be expressed as

$$Pattern(L_i) = intersection(C_j, Pattern(L_i - C_j)) \tag{10}$$

where C_j is an arbitrary cluster set in the list L_i; if L_i contains only one cluster set C_j, then *Pattern*(L_i) will return C_j. In order to find all frequent clusters that appear i times in L, it is necessary to obtain the combination set, given by

$$L^i_{com} = \{L^k_i | k = 1, 2, \ldots, \binom{|L|}{i}\} \tag{11}$$

where L^i_{com} includes all combinations that contain i cluster sets of L; the number of combinations is $\binom{|L|}{i}$. Then, the frequent cluster set can be obtained as

$$FC_i = \bigcup\nolimits_{l \in L^i_{com}} Pattern(l) \tag{12}$$

where FC_i contains all frequent clusters whose support is i. Then, all frequent clusters that appear more than Sup_{min} times can be obtained using the following algorithm (Box 3):

Multi-clustering strategy. As discussed in Section 3.2.1, the smaller the value of the constraint ω, the more similar are the traffic characteristics of the road segments in the same cluster; this may improve the estimation accuracy. However, the frequent cluster set covers fewer road segments because of the more stringent constraint. To resolve this conflict, a multiple-clustering strategy is adopted. Multiple constraints $\omega_1, \omega_2, \ldots \omega_m$ are selected for clustering and pattern mining; then, a list of frequent cluster sets, $FCL = \{FC_\omega | \omega = \omega_1, \omega_2, \ldots, \omega_m\}$, is generated, where FC_ω is the frequent cluster set corresponding to the constraint ω. In the process of

Box 3. Traffic frequent pattern mining algorithm.

```
Algorithm 3 TrafficPatternMining: Traffic frequent pattern mining
Input: L = {C_1, C_2, ..., C_D}; Sup_min: minimum support
Output: FC = {FC_i | i = Sup_min, Sup_min +1, ..., D};
1: FC ← ∅;
2: for i ← Sup_min to D
3:   FC_i ← ∅;
4:   L_com ← i-combinations from L;
5:   FC_i ← Get all frequent clusters from (12);
6:   FC ← FC ∪ {FC_i};
7: end for
8: return FC;
```

finding the cluster that contains the road segment r, the frequent cluster set with smaller ω should be considered first.

Probability calculation

Emission probability. For a road segment r that belongs to the frequent cluster C, its traffic condition $x^{r,t}$ at time slot t approximates the traffic conditions $y^{r,t}$ of other road segments in C. Thus, the difference *diff* between $x^{r,t}$ and $y^{r,t}$ can be adopted to calculate the emission probability $Pr(y^{r,t}|x^{r,t})$. According to observations, the emission probability follows an exponential distribution; hence, it can be calculated as

$$Pr(y^{r,t}|x^{r,t}) = \lambda e^{-\lambda \cdot diff} = \lambda e^{-\lambda \frac{1}{|C'|}\sum_{i \in C'} |x^{r,t} - x^{i,t}|} \tag{13}$$

where, λ is the parameter of the exponential distribution ($\lambda > 0$) and *diff* is the average traffic condition difference between the road segment r and road segments in $C' = \{r' \in C | \, r' \neq r\}$.

In order to find the appropriate frequent cluster C, the frequent cluster set FC_ω with a smaller constraint value ω should be considered preferentially, and in the frequent cluster set $FC = \{FC_i \,|i = Sup_{min}, Sup_{min} + 1, \ldots, D\}$, the clusters that appear more frequently should be considered first.

Transition probability. Through data analysis, in a relatively short time interval, the traffic condition at time slot $t+1$ is close to that at time slot t. Thus, the traffic state change $\Delta x = |x^{r,t} — x^{r,t+1}|$ is employed to measure the state transition. According to observations, the state transition probability follows an exponential distribution, and it can be expressed as

$$Pr(x^{r,t}, x^{r,t+1}) = \beta e^{-\beta \cdot \Delta x} \tag{14}$$

where, β is the parameter of the exponential distribution ($\beta > 0$).

Candidate selection

As discussed in Sections 3.2 and 3.3, the state $x^{r,t+1}$ may approximate the previous state $x^{r,t}$ and the observations $y^{r,t+1} = \{x^{i,t+1} | i \in C \text{ and } i \neq r\}$, where C is the frequent cluster that contains the road segment r. Then, the value range of $x^{r,t+1}$ is set as $[x_{min}\text{-}\mu, x_{max}\text{+}\mu]$, where $x_{min} = min(\{x^{r,t}\} \cup y^{r,t+1})$, $x_{max} = max(\{x^{r,t}\} \cup y^{r,t+1})$, and μ is used to avoid missing valid values. For computational convenience, the range should be discretized to finite candidates denoted by the

set

$$CS = \left\{ x_i \middle| x_i = x_{min} - \mu + i \cdot \frac{x_{max} - x_{min} + 2\mu}{N_{cand} - 1}, i = 0, 1, 2 \ldots, N_{cand} - 1 \right\} \quad (15)$$

where N_{cand} is the number of candidates. In order to facilitate algorithm design, the candidate set is $CS = \{x^{r,t+1}\}$ when the state at time slot $t+1$ is obtained from the samples. Then, it is not necessary to find the missing state sub-sequences discussed in Section 2; the entire state sequence can be estimated in a single process.

Real-time algorithm

For the road segment r at time slot t, a list $PreSList = \{Se_i \,|i = 1,2,\ldots m\}$ is used to store previous surviving state sequences. A sequence is denoted by $Se = (SS, JP)$, where $SS = \{x^{r,1}, x^{r,2}, \ldots, x^{r,t-1}\}$ stores previous consecutive candidate states and JP is the joint probability. Using Algorithm 4 (Box 4), the current candidate sequence list $SList$ is obtained according to $PreSList$ and the candidate states CS_t of road segment r at time slot t. Then, the state sequence with the maximum joint probability in $SList$ is the optimal solution of road segment r at time slot t; the sequence is given by $argmax_{Se \in SList}\{Se.JP\}$. For the first state, the initial joint probability of the state sequence is the emission probability of the candidate states. Obviously, the algorithm can output the estimated states in real time; thus, it is applicable to online application.

Box 4. Real-time traffic estimation algorithm based on HMM.

Algorithm 4 TrafficEstimation: Real-time traffic estimation based on HMM

Input: *PreSList*: List of surviving sequences of road segment *r* at time slot *t-1*; *t*: time.

Output: *SList*: List of surviving sequences of road segment *r* at time slot *t*.

```
SList ← PreSList; //SList is a current surviving sequence list
CS_t ← Get candidate states; //Discussed in Section 3.4;
SListTemp ← ∅;
if t == 1
  for x in CS_t
    Construct a new state sequence Se; Set x as the starting state;
    Se.JP ← Pr(y^{r,t} | x); //Discussed in Section 3.2;
    Add Se into SList;
  end for
else
  for x in CS_t
    Se ← argmax_{Se∈SList}{ Se.JP·Pr(x^{r,t-1}, x)} ;
    Set x as the t-th state of Se;
    Se.JP ← Pr(y^{r,t} | x)·Se.JP·Pr(x^{r,t-1}, x);
    Add Se to the temporary list SListTemp;
  end for
  SList ← SListTemp;
end if
output argmax_{Se∈SList}{ Se.JP} ; //Real-time output current solution;
return SList;
```

Results and Discussion

For the experiments, 8559 arterial road segments were selected; the roads cover the main regions of central Beijing. The traffic conditions between 6:00 and 24:00 were considered, and the time was divided into 108 time slots at 10-min intervals (e.g., the first time slot was 6:00–6:10 and the 12th time slot was 7:50–8:00).

The taxi trajectory data in Beijing during November 2012 served as the FCD data, obtained from 12,600 taxis. The data samples of six weekdays were selected for a case study; five of these days were used for frequent cluster mining and parameter estimation, and the remaining day was used to test the estimation model. Before the experiments were performed, the trajectory data were matched to the road network using map-matching methods [34–36], and anomalous samples were eliminated.

The model was implemented using a Java platform on a computer having a quad-core CPU (2.2 GHz) and 8-GB memory.

Frequent cluster mining

Six constraint values $\{\omega \mid \omega = 10, 15, 20, 25, 30, 35\}$ were considered in the clustering analysis stage. The average weight w_{av} discussed in Section 3.2.1 was employed to measure the degree of similarity, which decreased as w_{av} increased. As shown in Table 1, the mean w_{av} of the cluster set and the average number of objects in each cluster, $ON_{average}$, increase with ω. Clusters having a single object cannot be used for estimation; the proportion of such clusters, r_{single}, decreases as ω increases. For traffic estimation, a perfect cluster set has small average w_{av}, large $ON_{average}$, and small r_{single}. A cluster set having small average w_{av} is more likely to have small $ON_{average}$ and large r_{single}, which confirms the existence of the contradiction discussed in Section 3.2.3. Therefore, it is necessary to adopt a multi-clustering strategy.

As shown in Fig 5, the traffic characteristics of the road segments in the same cluster have a very high degree of similarity when the average weight w_{av} is small, such as clusters a, b, and c. As the average weight w_{av} increases, the degree of similarity of the cluster decreases and the number of the objects in the cluster increases.

Samples of five days were selected for frequent cluster mining, and the minimum support Sup_{min} was set as 3. Table 2 lists the coverage rates of the frequent clusters, which is given by the ratio $R_{cover} = N_{cover}/N_{total}$, where N_{cover} is the number of road segments in the frequent cluster set and N_{total} is the total number of road segments. The coverage rate increases with ω, and the support of the most frequent clusters is less than or equal to 4. When ω increases to 35, the coverage rate of the frequent cluster set reaches 96.76%, which indicates that the set of these ω is sufficient and appropriate for this study.

The road segments that are adjacent to each other may have similar traffic characteristics; this property can be used instead of clustering for finding similar road segments. However, in contrast to our assumption, this is not very likely in practice. As shown in Fig 6, although the

Table 1. Accuracy and coverage of cluster sets corresponding to different ω.

ω	mean w_{av}	$ON_{average}$	r_{single}(%)
10	12.29	2.34	21.51
15	12.97	2.69	15.06
20	14.61	3.67	9.21
25	16.58	5.09	4.71
30	18.71	7.29	2.34
35	20.77	11.04	1.03

doi:10.1371/journal.pone.0145348.t001

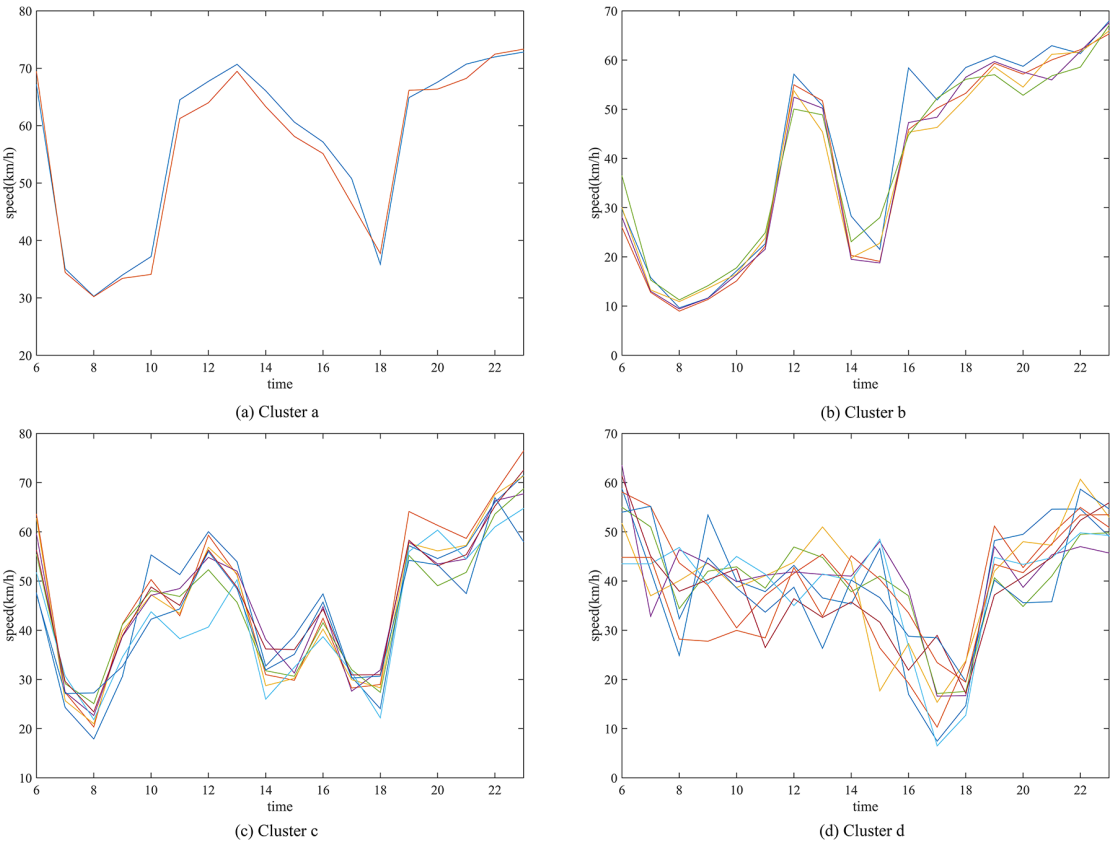


Fig 5. Traffic condition sequences of the road segments in four clusters. (a) The average weight of cluster a is 4.71, (b) the average weight of cluster b is 9.75, (c) the average weight of cluster c is 14.58, and (d) the average weight of cluster d is 24.93.

doi:10.1371/journal.pone.0145348.g005

proportion of road segments whose adjacent segments are in the same cluster increases with ω, this proportion is still low. Therefore, it is more reasonable to find similar road segments by clustering rather than by adjacency relationships.

Parameter estimation

Statistical analysis of the distribution of *diff* was carried out in order to estimate the parameter λ in (13). In different frequent cluster sets corresponding to specific values of ω, the distribution of *diff* is different. In order to observe the distribution of *diff*, we calculated the ratio of each *diff* value to the total number of samples. As shown in Fig 7, the steepness of the distribution curve increases with ω, which indicates that the road segments in the frequent cluster

Table 2. Coverage rate of the frequent cluster set for each ω (%).

ω/support	≥5	≥4	≥3
10	0.71	4.79	16.75
15	1.33	6.49	20.13
20	1.77	8.8	28.68
25	2.67	14.13	45.76
30	4.84	26.04	75.58
35	7.2	39.1	96.76

doi:10.1371/journal.pone.0145348.t002

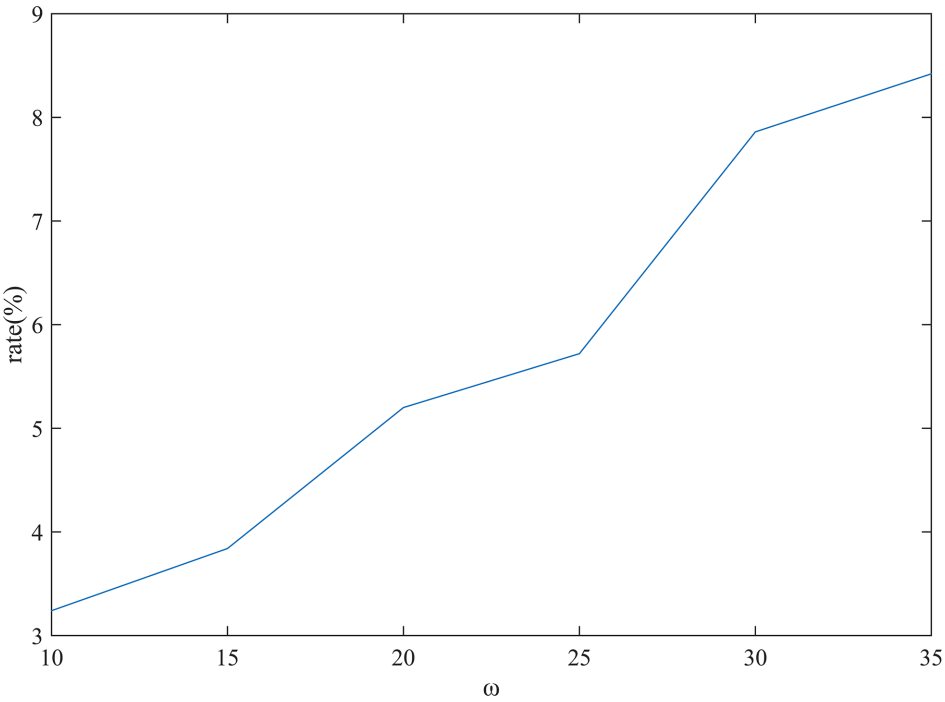


Fig 6. Proportion of road segments whose adjacent road segments are also in the same cluster.

doi:10.1371/journal.pone.0145348.g006

generated on the basis of a smaller ω are more likely to have a higher degree of similarity, because the probability that *diff* takes a smaller value is higher.

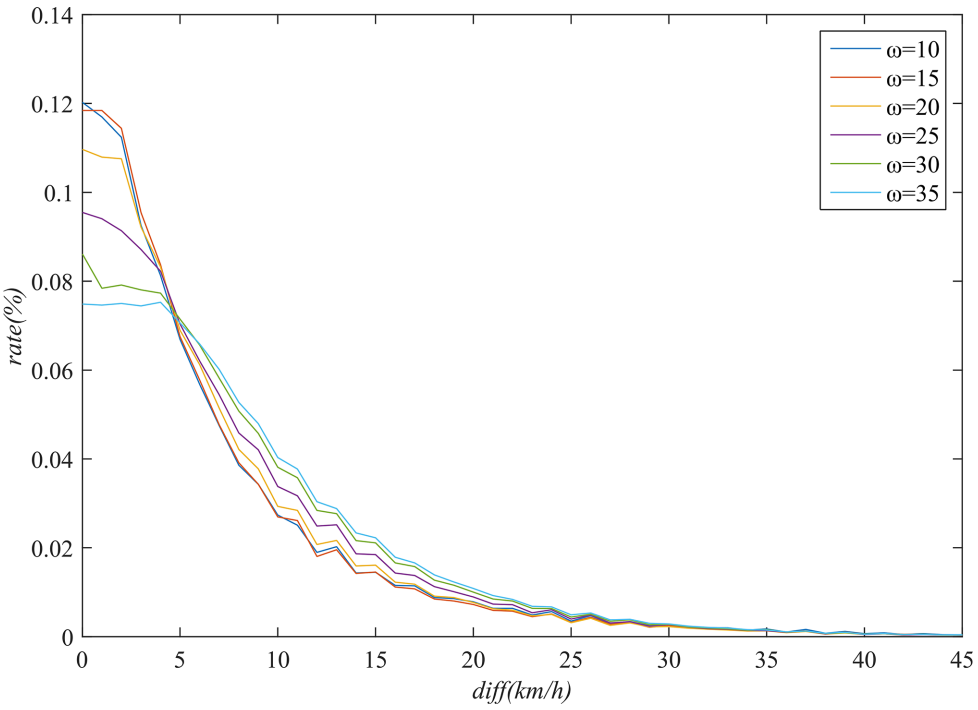


Fig 7. Distribution of diff in frequent cluster sets corresponding to different values of ω.

doi:10.1371/journal.pone.0145348.g007

Table 3. Estimated parameters of the emission probability equation corresponding to different frequent cluster sets.

ω	λ	SSE	R-square	RMSE
10	0.1337	0.000689	0.988	0.0030
15	0.1351	0.000847	0.986	0.0033
20	0.1251	0.000775	0.986	0.0032
25	0.1123	0.000694	0.986	0.0030
30	0.1012	0.000902	0. 980	0.0034
35	0.09267	0.001353	0. 966	0.0042

doi:10.1371/journal.pone.0145348.t003

The parameter λ was calculated as *1/E(diff)*, where *E(diff)* is the expectation of *diff*, and the equation was initialized with an initial parameter λ^*; then, the parameter was learned by iterative computation until it converged to a specific value. Table 3 lists λ values for six frequent cluster sets, $FCL = \{FC_\omega | \omega = 10, 15, 20, 25, 30, 35\}$, as well as the sum of squared errors (SSE), root-mean-square error (RMSE), and R-square, which indicate that the equation works well for the samples.

As shown in Fig 8, the traffic state change Δx follows an exponential distribution. The parameter β is calculated as *1/E(Δx)*, where *E(Δx)* is the expectation of Δx; after iterative computation, the estimated values of β, SSE, RMSE, and R-square are 0.09451, 0.000528, 0.002436, and 0.9871, respectively.

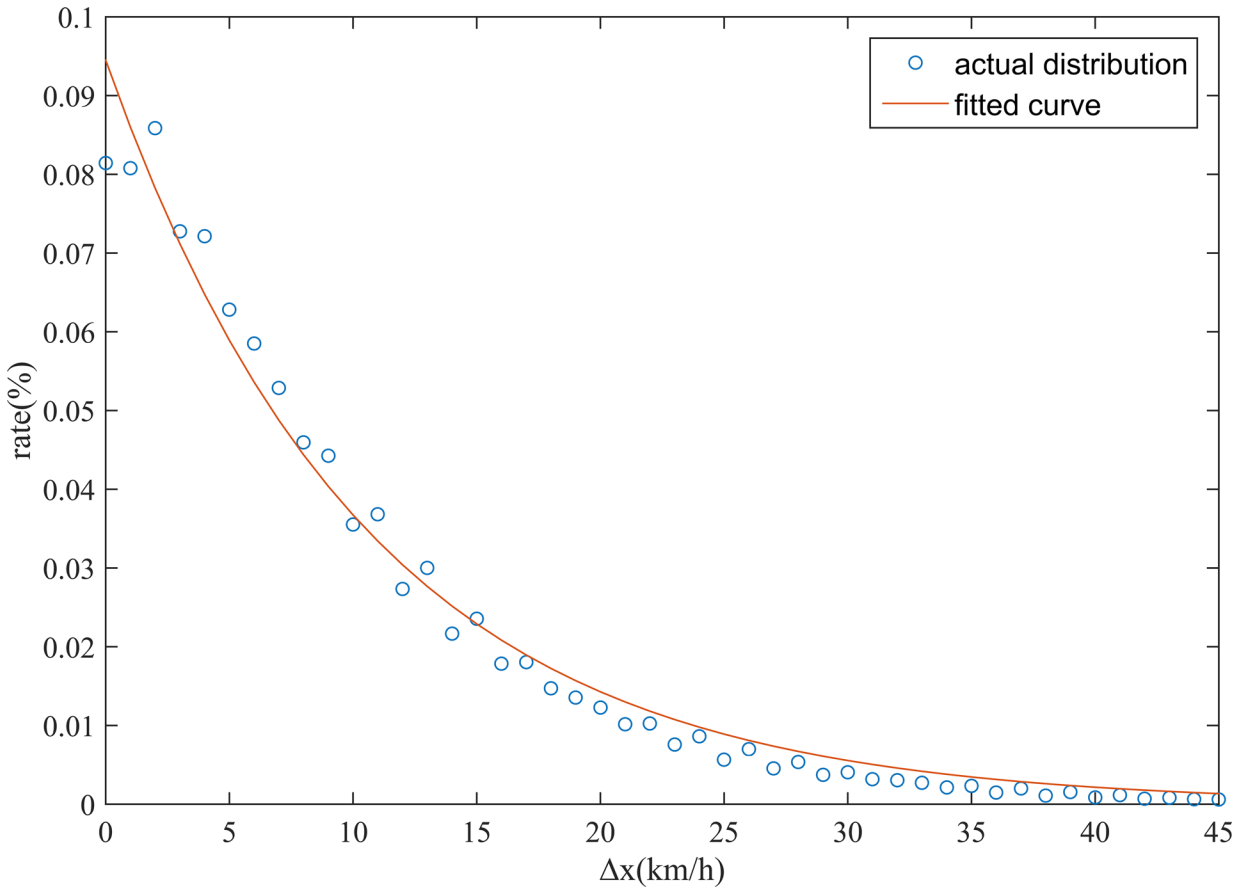


Fig 8. Distribution of the traffic state change Δx.

doi:10.1371/journal.pone.0145348.g008

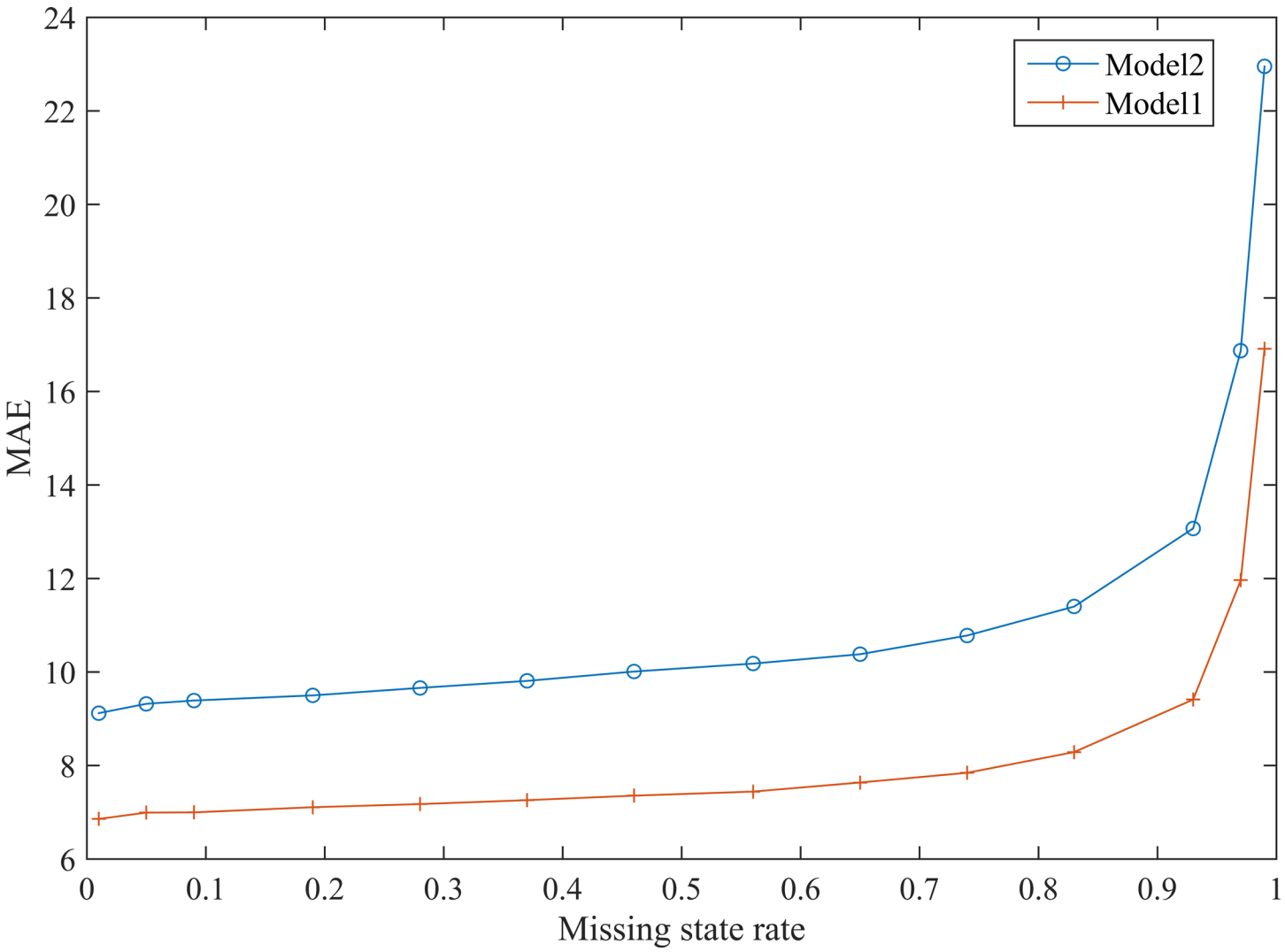


Fig 9. Comparison of accuracies of the two models.

doi:10.1371/journal.pone.0145348.g009

Model accuracy and efficiency

A state sequence set of 8559 arterial road segments was prepared for testing. Because it is difficult to obtain the complete state set of these road segments, the states that were actually obtained were considered for accuracy analysis; the total number of these states, N_{base}, was 6.19×10^5. Among these states, a number of states were randomly selected as the missing states that need to be estimated. The missing state rate is denoted by $R_{miss} = N_{miss}/N_{base}$, where N_{miss} is the number of missing states. The mean absolute error (MAE) was employed to measure the estimation accuracy, and it is given by

$$MAE = \frac{1}{N_{estim}} \sum_{i=1}^{N_{estim}} |\hat{x}_i - x_i| \tag{16}$$

where N_{estim} is the number of states estimated, $\hat{x}_i$ is the estimated value of the i-th state, and x_i is the true value of the i-th state.

Table 4. Estimation accuracy and coverage corresponding to different values of the missing state rate R_{miss}.

R_{miss}(%)	MAE(km/h)	R_{estim}(%)	R_{valid}(%)
0.93	6.86	0.92	99.99
4.63	6.99	4.61	99.98
9.26	7.01	9.2	99.94
18.52	7.11	18.24	99.72
27.78	7.18	26.77	98.99
37.04	7.26	35.88	98.84
46.3	7.36	44.93	98.63
55.56	7.44	53.92	98.36
64.81	7.64	62.8	97.99
74.07	7.84	71.51	97.44
83.33	8.29	79.41	96.08
92.59	9.41	85.12	92.53
97.22	11.96	82.05	84.83
99.07	16.91	64.58	65.51

doi:10.1371/journal.pone.0145348.t004

The accuracies of two models, Model 1 and Model 2, were compared. Model 1 is the proposed model, which finds road segments with similar traffic states via clustering and frequent pattern mining, whereas Model 2 assumes that adjacent road segments have similar traffic conditions. As shown in Fig 9, MAE increases with R_{miss}, and the MAE of Model 2 is significantly higher than that of Model 1, which implies that the proposed model is more accurate.

If there is no reference state, such as the previous state or the states of similar road segments, for the state $x^{r,t}$, the state cannot be estimated. The rate of the states that cannot be estimated is R_{miss}-R_{estim}; then, the number of states that can be estimated is $R_{valid} = 1$-(R_{miss}-R_{estim}), where $R_{estim} = N_{estim}/N_{base}$. As shown in Table 4, MAE increases gradually before R_{miss} reaches 83.33%, and R_{valid} remains high until R_{miss} reaches 92.59%, which indicates that the model is applicable to very sparse sample data.

The cumulative distribution function (CDF) of the estimation error E is given by

$$F_E(e) = P(E \leq e) = \frac{N(E \leq e)}{N_{estim}} \tag{17}$$

where estimation error E is the absolute value of the difference between the estimated and observed values, N_{estim} is the number of estimated states, and $N(E \leq e)$ is the number of estimated states whose error is less than or equal to e. Fig 10 shows the CDFs of estimation errors corresponding to different values of the missing state rate R_{miss}. Before R_{miss} reaches 83.33%, the CDF curve is steeper, which indicates that most errors are small. For example, when R_{miss} = 83.33%, more than 52.79% of the errors are less than or equal to 5 and more than 76.88% of the errors are less than or equal to 10.

The states that are obtained are considered as the true states; then, the estimated error distribution function in the global scope is given by

$$F'(e) = 1 - R_{miss} + F_E(e) \cdot R_{estim} \tag{18}$$

Table 5 summarizes the global distribution of the estimated error, which reflects the accuracy of the model corresponding to different values of the missing state rate R_{miss}. According to the error distribution, it is easy to determine whether the accuracy of the model meets the requirements of the application. For example, in an application that requires 90% of the errors

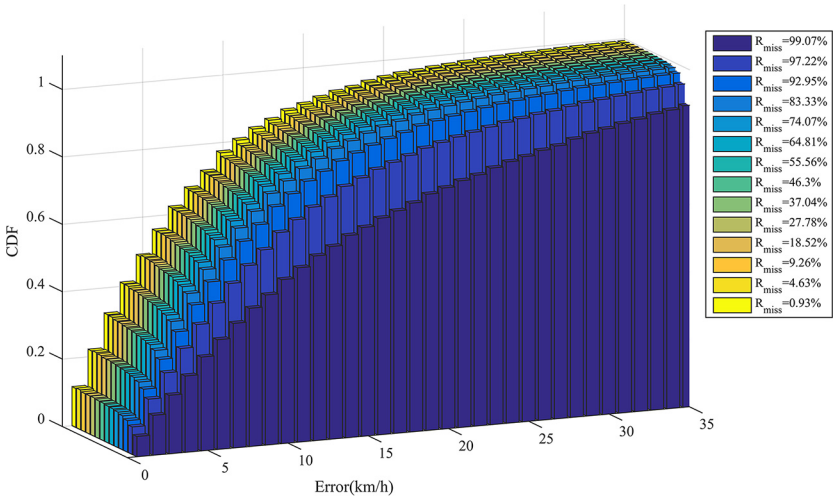


Fig 10. Estimation error CDF corresponding to different values of the missing state rate R_{miss}.

doi:10.1371/journal.pone.0145348.g010

are to be less than 5 km/h, if the missing state rate is less than 18.52%, then the model may be suitable for the application.

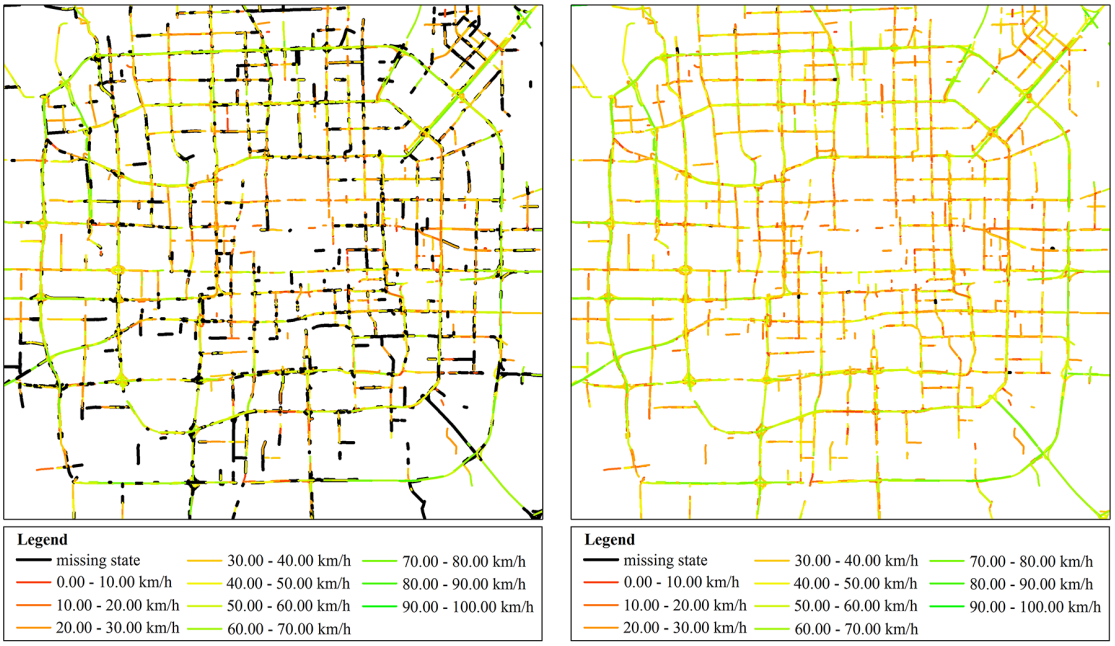
In our data source, the missing state rate of the arterial roads was around 33%, while the missing state rate of the other roads was around 65%. Samples of 8559 arterial roads were considered. The error was less than or equal to 5 (resp. 10) for more than 84.84% (resp. 92.61%) of the states; this indicates a high estimation accuracy. Fig 11(A) shows the traffic state map of the arterial roads in Beijing at the 50th time slot (14:10–14:20) before estimation, and Fig 11(B) shows the states of the roads after estimation. Most of the missing states were estimated, and the estimated values were very close to the true values.

The main factor that affects the efficiency of the model is the number of candidates for the hidden states, N_{cand}, which has been discussed in Section 3.4. Several values of N_{cand} were selected for the experiments, where the missing state rate was around 74%. The results show that the accuracy improved as N_{cand} increased; however, the time cost increased significantly

Table 5. Global cumulative distribution of the estimated error corresponding to different values of the missing state rate R_{miss}.

R_{miss}(%)	F'(5)(%)	F'(10) (%)	F'(20) (%)	F'(30) (%)
0.93	99.59	99.81	99.96	99.98
4.63	97.96	99.06	99.79	99.94
9.26	95.90	98.09	99.55	99.86
18.52	91.66	95.98	98.92	99.54
27.78	87.06	93.43	97.75	98.70
37.04	82.66	91.26	97.16	98.47
46.30	78.22	89.00	96.45	98.12
55.56	73.71	86.61	95.66	97.68
64.81	68.84	83.9	94.64	97.10
74.07	63.68	80.91	93.32	96.28
83.33	57.43	76.58	90.79	94.41
92.59	48.53	68.92	84.88	89.67
97.22	37.74	56.32	72.91	79.33
99.07	21.94	34.32	48.19	56.34

doi:10.1371/journal.pone.0145348.t005



(a) Before estimation (b) After estimation

Fig 11. Traffic condition of arterials in Beijing at 14:20–14:30. (a) Traffic condition map before estimation, and (b) traffic condition map after estimation; the black regions represent missing states.

doi:10.1371/journal.pone.0145348.g011

Table 6. Efficiency of the model corresponding to different values of N_{cand}.

N_{cand}	MAE (km/h)	cost (s)	speed (states/s)
4	8.35	21	19456.38
6	7.9	25	16343.36
8	7.82	52	7857.38
10	7.78	66	6190.66
12	7.76	98	4169.22
14	7.76	130	3142.95
16	7.76	154	2653.14
18	7.76	213	1918.23
20	7.76	243	1681.41

doi:10.1371/journal.pone.0145348.t006

(Table 6). When N_{cand} reached around 12, the accuracy stabilized and the model could estimate approximately 4169.22 states per second. From the viewpoint of practical application, the model can meet the efficiency requirements of metropolitan real-time traffic estimation.

Conclusion

This paper presented an effective and efficient HMM-based model for urban-scale traffic estimation using floating car data. Clustering analysis and pattern mining were adopted to analyze a large data set of real probe data collected from a fleet of 12,600 taxis in Beijing, China, and it was found that there exist frequent clusters in which the road segments have similar traffic characteristics. Comparative analysis showed that the model based on clustering is more effective than the model based on adjacency relationships for traffic estimation. In order to achieve a trade-off between clustering accuracy and coverage, a multi-clustering strategy was adopted

in the estimation process. Experimental results showed that the model can be applied to different scenarios; even when more than 70% of the original data are missing, the model can guarantee that more than 80% of the states have relatively small errors. In addition, the model was implemented using a real-time algorithm, which offers higher precision and has a broader scope for application than some offline traffic estimation algorithms.

Author Contributions

Conceived and designed the experiments: XW LP TC. Performed the experiments: XW ML XY JS. Analyzed the data: XW ML XY JS. Contributed reagents/materials/analysis tools: XW ML XY JS. Wrote the paper: XW ML XY JS.

References

1. Kong QJ, Zhao QK, Wei C, Liu YC. Efficient Traffic State Estimation for Large-Scale Urban Road Networks. Ieee T Intell Transp. 2013 Mar; 14(1):398–407.
2. Leontiadis I, Marfia G, Mack D, Pau G, Mascolo C, Gerla M. On the Effectiveness of an Opportunistic Traffic Management System for Vehicular Networks. Intelligent Transportation Systems, IEEE Transactions on. 2011; 12(4):1537–48.
3. Yeon J, Elefteriadou L, Lawphongpanich S. Travel time estimation on a freeway using Discrete Time Markov Chains. Transportation Research Part B: Methodological. 2008; 42(4):325–38.
4. Bramberger M, Brunner J, Rinner B, Schwabach H. Real-time video analysis on an embedded smart camera for traffic surveillance. Real-Time and Embedded Technology and Applications Symposium, 2004 Proceedings RTAS 2004 10th IEEE; 2004 May 25–28; 2004. p. 174–181.
5. Herrera JC, Bayen AM. Traffic Flow Reconstruction Using Mobile Sensors and Loop Detector Data. TRB 87th Annual Meeting Compendium; 2007.
6. Hiribarren G, Herrera JC. Real time traffic states estimation on arterials based on trajectory data. Transport Res B-Meth. 2014 Nov; 69:19–30.
7. Herrera JC, Work DB, Herring R, Ban X, Jacobson Q, Bayen AM. Evaluation of traffic data obtained via GPS-enabled mobile phones: The Mobile Century field experiment. Transportation Research Part C: Emerging Technologies. 2010; 18(4):568–83.
8. de Fabritiis C, Ragona R, Valenti G. Traffic Estimation And Prediction Based On Real Time Floating Car Data. Intelligent Transportation Systems, 2008 ITSC 2008 11th International IEEE Conference on; 2008 Oct 12–15; 2008. p. 197–203.
9. Bar-Gera H. Evaluation of a cellular phone-based system for measurements of traffic speeds and travel times: A case study from Israel. Transportation Research Part C: Emerging Technologies. 2007; 15 (6):380–91.
10. Breitenberger S, Grueber B, Neuherz M, Kates R. Traffic information potential and necessary penetration rates. Traffic Engineering & Control. 2004; 45(11):396–401.
11. Vandenberghe W, Vanhauwaert E, Verbrugge S, Moerman I, Demeester P. Feasibility of expanding traffic monitoring systems with floating car data technology. Iet Intell Transp Sy. 2012 Dec; 6(4):347–54.
12. Herring R, Hofleitner A, Abbeel P, Bayen A. Estimating arterial traffic conditions using sparse probe data. Intelligent Transportation Systems (ITSC), 2010 13th International IEEE Conference on; 2010 Sept 19–22; 2010. p. 929–36.
13. Zhu YM, Li Z, Zhu HZ, Li ML, Zhang Q. A Compressive Sensing Approach to Urban Traffic Estimation with Probe Vehicles. Ieee T Mobile Comput. 2013 Nov; 12(11):2289–302.
14. Bejan AI, Gibbens RJ. Evaluation of velocity fields via sparse bus probe data in urban areas. Intelligent Transportation Systems (ITSC), 2011 14th International IEEE Conference on; 2011 Oct 5–7; 2011. p. 746–53.
15. Seo T, Kusakabe T, Asakura Y. Estimation of flow and density using probe vehicles with spacing measurement equipment. Transport Res C-Emer. 2015 Apr; 53:134–50.
16. Fowe AJ, Chan YP. A microstate spatial-inference model for network-traffic estimation. Transport Res C-Emer. 2013 Nov; 36:245–60.
17. Ramezani M, Geroliminis N. Queue Profile Estimation in Congested Urban Networks with Probe Data. Comput-Aided Civ Inf. 2015 Jun; 30(6):414–32.

18. Tang TQ, Shi WF, Shang HY, Wang YP. An extended car-following model with consideration of the reliability of inter-vehicle communication. Measurement. 2014; 58:286–293.
19. Gupta A.K.; Sharma S.; Redhu P. Analyses of lattice traffic flow model on a gradient highway. Commun Theor Phys. 2014; 62:393–404.
20. Gupta A.K.; Redhu P. Jamming transition of a two-dimensional traffic dynamics with consideration of optimal current difference. Phys Lett A. 2013; 377:2027–2033.
21. Gupta A.K. A section approach to a traffic flow model on networks. Int J Mod Phys C. 2013, 24.
22. Tang TQ, Li CY, Huang HJ. A new car-following model with the consideration of the driver's forecast effect. Physics Letters A. 2010; 374(38):3951–3956.
23. Yu SW, Shi ZK. Dynamics of connected cruise control systems considering velocity changes with memory feedback. Measurement. 2015; 64:34–48.
24. Ge J, Orosz G. Dynamics of connected vehicle systems with delayed acceleration feedback. Transportation Research Part C. 2014; 46:46–64.
25. Tang TQ, Huang HJ, Shang HY. A dynamic model for the heterogeneous traffic flow consisting of car, bicycle and pedestrian. International Journal of Modern Physics C. 2010; 21:159–176.
26. Yu SW, Shi ZK. An extended car-following model considering vehicular gap fluctuation. Measurement. 2015; 70:137–147.
27. Yu SW, Shi ZK. An extended car-following model at signalized intersections. Physica A, 2014; 407:152–159.
28. Yu SW, Shi ZK. An improved car-following model considering headway changes with memory. Physica A, 2015; 421:1–14.
29. Yu S, Liu Q, Li X. Full velocity difference and acceleration model for a car-following theory. Communications in Nonlinear Science & Numerical Simulation. 2013; 18(5):1229–1234.
30. Wang JW, Wang YS, Yun MP, Yang XG. Development of Urban Road Network Traffic State Dynamic Estimation Method. Math Probl Eng. 2015.
31. Work DB, Blandin S, Tossavainen O-P, Piccoli B, Bayen AM. A Traffic Model for Velocity Data Assimilation. Applied Mathematics Research eXpress. 2010 January 1; 2010(1):1–35.
32. Hofleitner A, Herring R, Abbeel P, Bayen A. Learning the Dynamics of Arterial Traffic From Probe Data Using a Dynamic Bayesian Network. Intelligent Transportation Systems, IEEE Transactions on. 2012; 13(4):1679–93.
33. von Luxburg U. A tutorial on spectral clustering. Stat Comput. 2007; 17(4):395–416.
34. Chen BY, Yuan H, Li QQ, Lam WHK, Shaw SL, Yan K. Map-matching algorithm for large-scale low-frequency floating car data. Int J Geogr Inf Sci. 2014 Jan 2; 28(1):22–38.
35. He ZC, She XW, Zhuang LJ, Nie PL. On-line map-matching framework for floating car data with low sampling rate in urban road networks. Iet Intell Transp Sy. 2013 Dec; 7(4):404–14.
36. Raymond R, Morimura T, Osogami T, Hirosue N. Map matching with Hidden Markov Model on sampled road network. International Conference on Pattern Recognition (ICPR); 11–15 Nov. 2012; Tsukuba: IEEE; 2012. p. 2242–5.