JOURNAL OF INFORMATION SCIENCE AND ENGINEERING 34, 781-799 (2018)
DOI: 10.6688/JISE.201805_34(3).0013

Urban Traffic Congestion Index Estimation With Open Ubiquitous Data*

MINGQI LV, YIFAN LI, TIEMING CHEN+ AND YINGLONG LI
College of Computer Science and Technology
Zhejiang University of Technology
Hangzhou, 310023 P.R. China
E-mail: {mingqilv; yifanli; tmchen; liyinglong}@zjut.edu.cn

Traffic congestion index (*i.e.*, TCI) is a metric for measuring urban road congestion degree. Since traffic congestion has been one of the major issues in most metropolises, it is a crucial demand to know the TCI of every road segment at every time slot. However, it is a challenging task, because the TCI sensors (*e.g.*, road-side sensors, floating vehicles, *etc.*) are limited in spatial or temporal dimension, leading to sparse TCI data in the spatial-temporal space. In this paper, we propose a method for estimating the missed TCI data for any road segment at any time slot, based on the sparse TCI data reported by the existing TCI sensors and a variety of open ubiquitous data. First, it extracts various urban features which have a correlation with the TCI data from the open ubiquitous data. Second, it fuses the urban features with the sparse TCI data using a collective matrix factorization algorithm, and collaboratively estimates the missed data. The advantage of our method is that it could adapt to the situation of high TCI data sparsity by incorporating external correlations from open ubiquitous data. We evaluate our method with extensive experiments based on a real-world TCI dataset and four open ubiquitous data sources. The results show the effectiveness of our method.

Keywords: traffic congestion index, open ubiquitous data, urban computing, collective matrix factorization, feature fusion

1. INTRODUCTION

Traffic congestion has been one of the major issues in most metropolises. Traffic congestion would cause many urban problems, *e.g.*, wasting energy, causing pollution, decreasing productivity, *etc.* TCI is a metric for quantifying the degree of urban road congestion [1], so the knowledge of TCI of every road segment at every time slot is valuable. For instance, a navigation system could leverage such knowledge to suggest the fastest path rather than the shortest path to drivers [2]. Such knowledge could also be used to support the decision making on the city's transportation planning [3].

Most big cities have deployed various monitoring systems to obtain TCI data. These TCI monitoring systems could be roughly divided into two categories according to the sensing techniques: static systems and dynamic systems. Static systems monitor TCI data based on data from stationary road-side sensors (*e.g.*, loop detector [4], visual camera [5], RFID reader [6], *etc.*), and dynamic systems are based on data generated by floating ve-

Received March 31, 2017; revised April 24 & May 12, 2017; accepted May 21, 2017.
Communicated by Shyi-Ming Chen.
* This work was supported by the Zhejiang Provincial Natural Science Foundation of China (No. LY18F020033), the Natural Science Foundation of China (Nos. 61772026, 61602412, 61502421), and the Joint Funds of the National Natural Science Foundation of China (No. U1509214).
+ The corresponding author.

hicles [7-9]. However, the results are usually sparse in spatial or temporal dimension due to the limitation of these systems. For static systems, they suffer from high deployment and maintenance costs. Thus, the expansion of their coverage is prohibitive, resulting in "*spatial sparsity*". For dynamic systems, although floating vehicles could travel to any road segment, their number is usually limited (buses or taxis). Thus, for a given road segment, only a small portion of time could have floating vehicles traveling on it, leading to "*temporal sparsity*".

To demonstrate the TCI data sparsity problem, we collected a real TCI dataset from a public traffic congestion monitoring website [10] that monitors TCI data of 199 road segments of Hangzhou city, China, based on stationary road-side sensors. Fig. 1 visualizes the monitored road segments. When zooming in the map, it can be found that a large proportion of road segments have not been monitored. It verifies the spatial sparsity of the TCI data. Besides, we analyzed the dataset and found that even for the monitored road segments, the TCI data only cover 65.16% time slots. Furthermore, if treating the TCI data of a given road segment on a given day as a unit, the units with complete data at every time slot only account for 8.55%. It verifies the temporal sparsity of the TCI data.

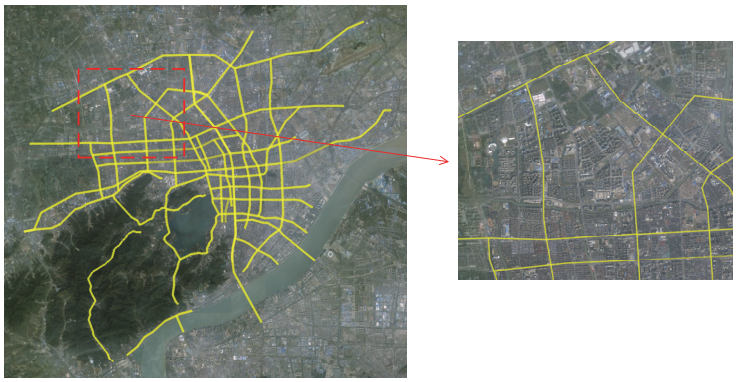


Fig. 1. The visualization of the monitored road segments.

Thus, it is crucial to have a method that could estimate the missed TCI data for any road segment at any time slot. According to the situations of data sparsity, we categorize the TCI data estimation into the following three problems.

- Data filling (estimating the missed TCI data of a road segment): The road segment has a certain amount of TCI data available, but misses TCI data at some time slots. It corresponds to the temporal sparsity situation.
- Data inference (estimating the unknown TCI data of a road segment): The road segment has not been monitored, and thus does not have any TCI data available. It corresponds to the spatial sparsity situation.
- Data prediction (estimating the future TCI data of a road segment): The historical TCI data of the road segment are available. It also corresponds to the temporal sparsity situation and is a special case of data filling. For a time slot, data filling problem misses TCI data on a part of road segments (it might be caused by temporary failure

of the sensors), while data prediction problem misses TCI data on all road segments (since the time slot is in the future).

The existing methods resolve the TCI data estimation problems by exploiting the internal spatial and temporal correlations of the TCI data [11, 12], so they would become less effective when the TCI data are highly sparse. Thus, a TCI data estimation method that could adapt to the situation of high TCI data sparsity is desirable.

In this paper, we borrow the idea from context-aware recommender systems [13], and propose a TCI data estimation method by incorporating external correlations from open data sources. First, we analyze the correlation between the TCI data and several open data sources (*i.e.*, road network, POIs, check-ins and geotagged photos). These open data sources can reflect the urban transportation development and human mobility patterns to a great extent, so they are suppose to have a correlation with the TCI data. For example, the traffic congestion pattern of a road segment within residential area would be greatly different from that of a road segment within commercial area. Likewise, a specific road segment is likely to embrace heavier traffic congestion during time slots with more user check-ins. Based on the analysis, we extract corresponding urban features. Second, since the extracted urban features reflect the potential correlations of TCI data, they could be used as "contexts" to enhance the modeling of spatial and temporal correlations of TCI data. Thus, we fuse them with the sparse TCI data based on a collective matrix factorization algorithm, and collaboratively estimate the unavailable TCI data.

The primary contributions of this paper are summarized as follows:

- We propose a method to address the spatial and temporal sparsity problem of urban TCI data by exploring four open ubiquitous data sources (*i.e.*, road network, POIs, check-ins and geotagged photos).
- We analyze the correlations between the four open ubiquitous data sources and the TCI data, and design various urban features that can capture these correlations from these data sources.
- We estimate the missed TCI data based on a collective matrix factorization algorithm, which exploits the correlations between the sparse TCI data and the extracted urban features.
- We evaluate our method using a real TCI dataset and several open ubiquitous datasets over a period of a year.

2. RELATED WORK

2.1 Traffic Condition Estimation

Conventional traffic monitoring systems use stationary road-side sensors (*e.g.*, inductive loop [4], visual camera [5], RFID reader [6], *etc.*) to observe traffic data (*e.g.*, speed, occupancy, volume, *etc.*). Then, traffic condition is calculated by learning a specific relation between these traffic data and the traffic conditions [14, 15]. Since road-side sensors often suffer from the problem of high installation and maintenance costs, some advanced methods use on-road vehicles as dynamic sensors and estimate traffic condition based on sensor data reported by the vehicles (*e.g.*, GPS [16, 17], cellular sig-

nal [18], acceleration [19], *etc.*). However, these methods treat each road segment independently, and do not take the correlation of traffic conditions between different road segments into account. On the other hand, some methods model several spatially connected road segments simultaneously [20-22]. However, these methods are still restricted to local road segments, and cannot to be applied to the entire road network.

Recently, more methods have turned to mine city-scale traffic condition patterns using massive trajectory data, generated by floating cars or mobile phone users. For example, Gonzalez *et al.* [23] proposed an adaptive fastest path algorithm based on traffic patterns mined from a large set of trajectory data. Wang *et al.* [24] investigated the road usage based on hidden road usage patterns mined from large-scale mobile phone data with detailed GIS data. Janecek *et al.* [25] inferred road congestion patterns using anonymous signaling data collected from a mobile cellular network. Chawla *et al.* [26] inferred the root cause of traffic anomalies by mining the GPS trajectories from taxicabs. However, these patterns are high-level abstraction of urban traffic conditions, ignoring the fine-grained information (*i.e.*, the traffic condition on any road segment at any time).

To estimate the fine-grained traffic information, some methods model the traffic conditions in a road network with a road-time matrix, where each entry denotes the traffic condition on a specific road segment at a particular time slot. Then, the missed entries in the matrix are estimated based on algorithms such as compressive sensing [11, 12]. However, these methods only consider the internal spatial and temporal correlations of traffic condition data. Unlike them, our method incorporates external correlations from other data sources (*i.e.*, spatial correlation from road network and POI datasets, temporal correlation from check-in and geotagged photo datasets). These external correlations would be greatly helpful when the original traffic condition data are extremely sparse.

2.2 Urban Computing

Urban computing is becoming a hot research area as a variety of urban data in urban spaces are collected [27]. The urban data implies rich knowledge about a city, which can be leveraged to help tackle the challenges of the city. For example, Majid *et al.* [28] designed an urban travel recommendation system based on the knowledge mined from online geotagged photo data and meteorological data. Zheng *et al.* [29] inferred the real-time air quality information throughout a city based on the air quality data reported by existing monitor stations and several urban data sources (*e.g.*, taxi trajectory data, meteorological data, *etc.*). Calegari and Celino [30] predicted the Milano city's land use and demographics based on heterogeneous open datasets (*e.g.*, POIs, call data records, *etc.*). Wilke and Portmann [31] introduced a collaborative urban planning use case in a cognitive city environment by using granular computing technique. Unlike these works, we propose a method to estimate the fine-grained TCI data leveraging open ubiquitous datasets (*i.e.*, road network, POIs, check-ins, and geotagged photos).

2.3 Context-Aware Recommender Systems

Our method is inspired by the idea of context-aware recommender systems [13], which take additional data such as user attributes (*e.g.*, gender, age, *etc.*), item attributes (*e.g.*, genres, keywords, *etc.*) or current situations (*e.g.*, current location, current mood,

etc.) into account. Such additional information is called "context" and can improve the accuracy of user-item rating prediction [32, 33]. Collective matrix factorization (or called context-aware matrix factorization) is currently the most accurate technique for context-aware recommender systems [34, 35]. Besides online recommendations, collective matrix factorization has also been widely used to conduct recommendations in urban environment [36, 37]. Our work adopts the collective matrix factorization technique to conduct TCI data estimation.

3. OVERVIEW

3.1 Preliminary

Definition 1 (Traffic Congestion Index): TCI is a numeric value to indicate how congested the road segment is currently. How to compute TCI is beyond the scope of this paper. As the TCI value increases, the road segment experiences heavier traffic congestion. Besides, traffic congestion is not an instant state, so a TCI value is associated with a road segment and a time slot. Thus, a TCI is a four-tuple $tci = (s, r, t_s, t_e)$, where s is the numeric value denoting the congestion level, r is the road segment, t_s to t_e is the time slot.

Definition 2 (Road Network): A road network contains a set of road segment, each of which has two terminal points and a series of intermediate points between the two terminal points. The road network dataset is downloaded from OpenStreetMap [38].

Definition 3 (POI): A POI (Point of Interest) is a place (*e.g.*, a restaurant, a shop, *etc.*) in the physical world, having attributes such as name, location and category. The POI dataset is created based on BaiduMap [39].

Definition 4 (Check-in): Some location-based social networking services allow a user to mark a place when the user arrives, known as a check-in. Each check-in has a location and a timestamp. The check-in dataset is collected from Weibo [40].

Definition 5 (Geotagged Photo): Some location-based social networking services allow a user to upload a photo, associated with the user's current location, known as a geotagged photo. Each geotagged photo has a location and a timestamp. The geotagged photo dataset is collected from Flickr [41].

3.2 Framework

Fig. 2 gives an overview of the architecture of our method. First, it extracts urban features from various open ubiquitous data sources (including road networks, POIs, check-ins and geotagged photos). These urban features are supposed to capture the potential correlations of TCI data. Second, it uses matrices to model the sparse TCI data and the urban features. Third, it collectively factorizes the sparse TCI data matrix with the aid of the dense urban feature matrices, and then resolves the three data estimation problems (*i.e.*, data filling, data inference and data prediction) by recovering the sparse TCI data matrix based on the factorized latent factors.

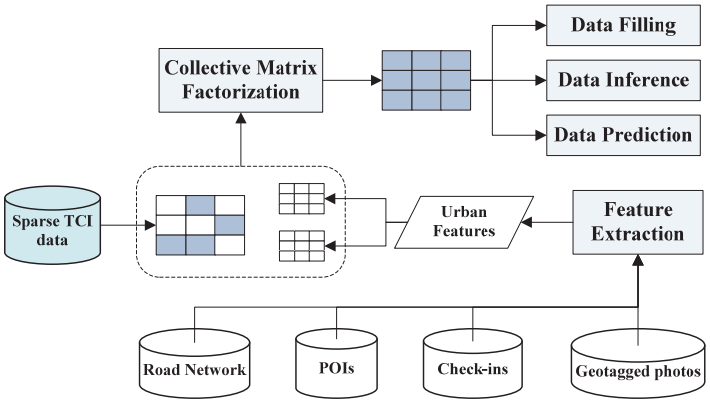


Fig. 2. The architecture of our method.

4. URBAN FEATURE EXTRACTION

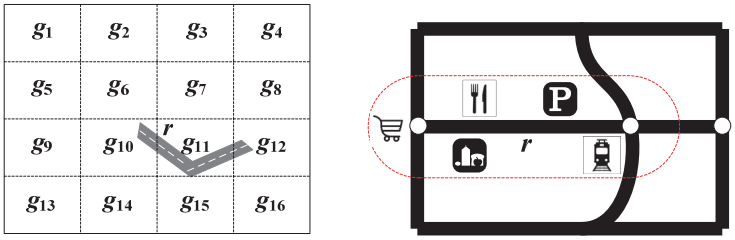
We extract two categories of urban features from the open ubiquitous data sources: geographic features and mobility features. Geographic features are extracted from road network and POI datasets, describing the static spatial contexts of a road segment. Mobility features are extracted from check-in and geotagged photo datasets, revealing the dynamic correlations between road segments from temporal aspect.

4.1 Geographic Features

4.1.1 Road network features

First, the objects of TCI data estimation are road segments in road network, so the basic features of road segments have a correlation with the TCI data. Given a road segment r, we extract the following five basic features as a feature vector f_r from it: (a) *road level* (indicating the road segment is urban express way, arterial road or small street); (b) *road direction* (indicating the road segment is a one-way road or not); (c) *number of intersections*; (d) *road length* (which is the total distance between all the consecutive points of r); and (e) *road tortuosity* (which is the ratio between road length and the Euclidian distance of the two terminal points of r).

Second, congested road segments are usually spatially close [21], so we calculate the spatial closeness between road segments as an extra road network feature. A direct way is to calculate the Euclidean distance between the centroids of all pairs of road segments. However, it would cause every road segment to maintain the distance with all other ones, leading to an over large feature vector. Thus, we calculate a coarse-grained feature vector f_c for a road segment r, indicating which part of the city it falls in. We partition the city into $N \times N$ grids, and the f_c of r is represented by the grid that r falls in and its eight neighbors. Note that we do not use the single grid that r falls in, because it cannot distinguish grids that are close to each other from grids that are far away from each other. For instance, as shown in Fig. 3 (a), f_c of road segment r is (0, 0, 0, 0, 0, 1, 1, 1, 0, 1, 1, 1, 0, 1, 1, 1). Thus, road segments that fall in closer grid would have more similar f_c.



(a) (b)

Fig. 3. Geographic features: (a) the coarse-grained spatial closeness of road segments; (b) the POIs around a road segment.

4.1.2 POI features

The category of POIs and their density near a road segment r indicate the land use of the region around r, which further influences the traffic patterns in the region. Fig. 4 gives an example to show the TCI at different time slots of six road segments on a typical weekday. Road segments A to C are picked from residential area of Hangzhou city, and road segments D to F are picked from scenic area. It can be found that road segments from residential area have fairly different traffic patterns with those from scenic area. Road segments A to C encounter high traffic congestion during rush-hours, and D to F have relatively high traffic congestion in the afternoon. Besides, TCI of road segments from residential area is generally higher than that of road segments from scenic area.

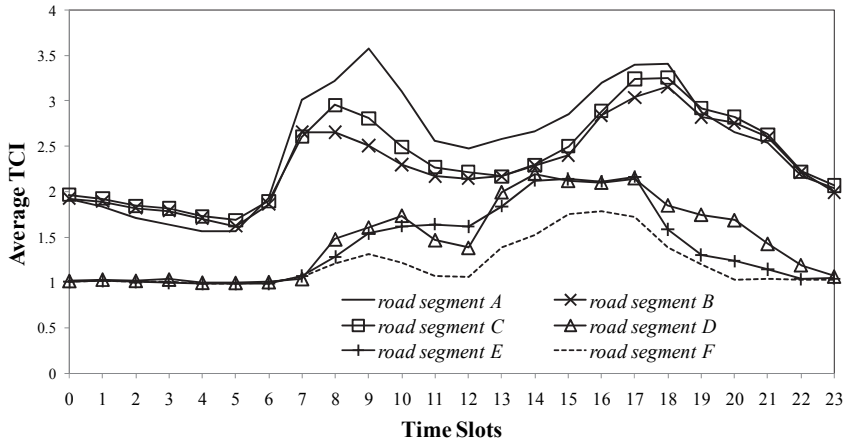


Fig. 4. The traffic patterns of road segments in different functional regions.

For a road segment r, from the POIs around r (as shown in Fig. 3 (b)), we calculate the density of POIs as a feature vector f_p across 8 categories, *i.e.*, *Residence*, *Work* (*e.g.*, companies, office buildings, *etc.*), *Commerce* (*e.g.*, supermarkets, malls, shops, *etc.*), *Restaurant*, *School*, *Transportation* (*e.g.*, railway station, subway station, airport, *etc.*), *Entertainment* (*e.g.*, cinema, bar, *etc.*), and *Scenery* (*e.g.*, park, lake, temple, *etc.*). The

density of POI category i around r is calculated based on TF-IDF as follow, where n_i^r is the number of POIs of category i around r, n^r is the total number of POIs around r, N_i is the total number of POIs of category i in the POI dataset, and N is the total number of POIs in the POI dataset.

$$f_i = \frac{n_i^r}{n^r} \times \log \frac{N}{N_i} \tag{1}$$

4.2 Mobility Features

Check-in and geotagged photo datasets from location-based social networks reflect different aspects of human mobility in a city. Check-in data usually reflect the mobility of local inhabitants, while geotagged photo data usually reflect the mobility of tourists [42]. Thus, a road segment with more check-ins is likely to be within an area with more popular local POIs (*e.g.*, restaurants, malls, *etc.*), while a road segment with more geotagged photos is likely to be within an area with more hot scenic spots. Obviously, this kind of human mobility is relevant to traffic congestion. In order to verify this fact, we pick two road segments (road segment A is from the scenic area and B is from the downtown area) and draw the TCI, the check-in number and the geotagged photo number on each hour of a day in Fig. 5 (the numbers are normalized into a value falling in [0, 1]). We can find that the check-in number has a strong correlation with the TCI (the Pearson correlation of A is 0.914 and the Pearson correlation of B is 0.865). Likewise, the geotagged photo number also has a positive correlation with the TCI (the Pearson correlation of A is 0.861 and the Pearson correlation of B is 0.738). The correlation between geotagged photo number and TCI is weaker than that between check-in number and TCI. It might be because that few photos are taken in the evening, even if the number of tourists is large.

For mobility features, given a road segment r, we calculate the average check-in and geotagged photo number around r for every time slot of a day over a long period.

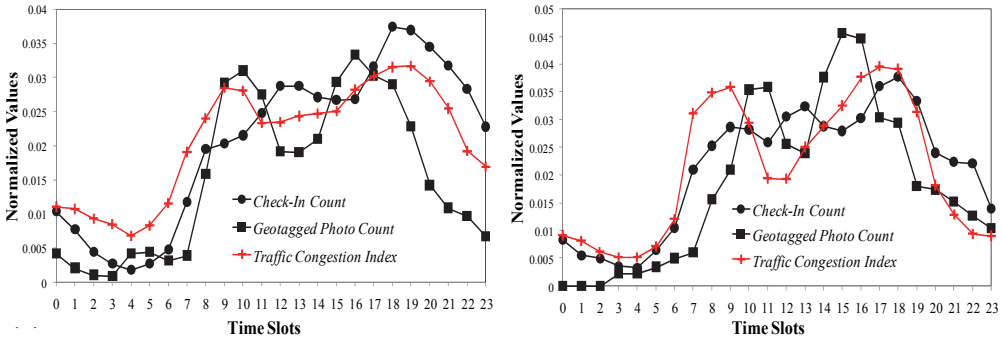


Fig. 5. The TCI, the number of check-ins, and the number of geotagged photos on each hour of a day: (a) road segment A from the scenic area; (b) road segment B from the downtown area.

5. TRAFFIC CONGESTION INDEX ESTIMATION

5.1 Matrix Construction

As shown in Fig. 6, we use matrices to model TCI and urban features as follows.

- Traffic congestion index: The TCI data could be modeled by using a matrix, where the first dimension denotes road segments $\boldsymbol{r} = [r_1, r_2, ..., r_J]$, and the second dimension denotes time slots of a day $\boldsymbol{t} = [t_1, t_2, ..., t_K]$, where each time slot lasts for an equal period of time. Since the TCI data are collected from a long period of time, we use two matrices $\boldsymbol{C}$ and $\boldsymbol{C'}$. $\boldsymbol{C}$ is built from the data of the current day, on which the estimation is performed. $\boldsymbol{C'}$ is built from the historical data, corresponding to the same time slots with $\boldsymbol{C}$. Thus, $\boldsymbol{C}(i, j)$ stores the TCI of road segment r_i at time slot t_j on the current day and $\boldsymbol{C'}(i, j)$ stores the average TCI of road segment r_i at time slot t_j over the historical days.
- Geographic features: For each road segment, we combine f_r, f_c and f_p as a long feature vector, and then place all the long feature vectors into a matrix $\boldsymbol{G}$, where each row denotes a road segment and each column represents a kind of feature, including basic road segment features, the spatial closeness of road segments, and the density of POIs around road segments.
- Mobility features: We use two matrices $\boldsymbol{M}$ and $\boldsymbol{M'}$ to model the mobility features, where each row denotes a road segment and each column denotes a time slot of a day $\boldsymbol{t} = [t_1, t_2, ..., t_L]$. $\boldsymbol{M}(i, j)$ and $\boldsymbol{M'}(i, j)$ contain the historical check-in number and geotagged photo number of road segment r_i at time slot t_j, respectively. Note that the time slot of mobility features is usually coarser than that of TCI (*i.e.*, $K > L$). It is because that the number of check-ins and geotagged photos is usually very limited (especially the geotagged photos). Thus, if we use too fine-grained time slots for mobility features (*i.e.*, setting a large L), a large amount of time slots would become "empty" (*i.e.*, no check-in or geotagged photo would be in these time slots).

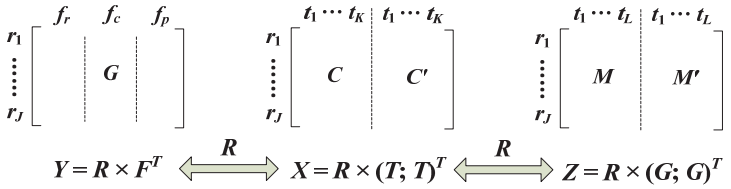


Fig. 6. An illustration of our method based on collective matrix factorization.

5.2 Collective Matrix Factorization

We uniformly formulate all the three TCI data estimation problems as a collaborative filtering problem, *i.e.*, to estimate the missed entries of $\boldsymbol{C}$. Although we could solve this problem by solely factorizing $\boldsymbol{C}$ based on the non-zero entries of it, the results would be not accurate enough. The reason is that some dimensions of $\boldsymbol{C}$ might be over sparse, and thus it is very difficult to capture the correlations between these sparse dimensions

and other dense dimensions. For example, in the data inference problem, the data are completely missed on some road segments, so it is very difficult to infer the correlations between these road segments and other ones that have available data.

To achieve a higher accuracy of estimation, we factorize $\boldsymbol{C}$ with the aid of $\boldsymbol{C'}$, $\boldsymbol{G}$, $\boldsymbol{M}$ and $\boldsymbol{M'}$, where $\boldsymbol{C'}$ contains the historical TCI patterns of every road segment, $\boldsymbol{G}$ models the correlations between road segments from spatial aspect, and $\boldsymbol{M}$ and $\boldsymbol{M'}$ model the correlations between road segments from temporal aspect. Fig. 6 illustrates our method based on collective matrix factorization, where three matrices $\boldsymbol{X}$, $\boldsymbol{Y}$ and $\boldsymbol{Z}$ are formulated based on $\boldsymbol{C}$, $\boldsymbol{C'}$, $\boldsymbol{G}$, $\boldsymbol{M}$ and $\boldsymbol{M'}$, *i.e.*, $\boldsymbol{X} = \boldsymbol{C}||\boldsymbol{C'}$, $\boldsymbol{Y} = \boldsymbol{G}$, $\boldsymbol{Z} = \boldsymbol{M}||\boldsymbol{M'}$. Then, we can factorize $\boldsymbol{X}$, $\boldsymbol{Y}$ and $\boldsymbol{Z}$ as: $\boldsymbol{X} \approx \boldsymbol{R} \times (\boldsymbol{T}; \boldsymbol{T})^T$, $\boldsymbol{Y} \approx \boldsymbol{R} \times \boldsymbol{F}^T$, $\boldsymbol{Z} \approx \boldsymbol{R} \times (\boldsymbol{G}; \boldsymbol{G})^T$, where $\boldsymbol{R}$, $\boldsymbol{T}$, $\boldsymbol{F}$ and $\boldsymbol{G}$ are low-rank matrices representing latent factors, $\boldsymbol{T}^T$ means the transpose of matrix $\boldsymbol{T}$. $\boldsymbol{X}$ and $\boldsymbol{Y}$ share latent factor $\boldsymbol{R}$. $\boldsymbol{X}$ and $\boldsymbol{Z}$ also share latent factor $\boldsymbol{R}$. The temporal dimensions of $\boldsymbol{X}$ and $\boldsymbol{Z}$ are factorized by different matrices (*i.e.*, $\boldsymbol{T}$ and $\boldsymbol{G}$, respectively), because they have different granularity on time slots. Since $\boldsymbol{Y}$ and $\boldsymbol{Z}$ are dense matrices, they can effectively improve the accuracy of inferring the correlations between different dimensions of $\boldsymbol{X}$, if we factorize them with $\boldsymbol{X}$ collaboratively.

The objective function of our collective matrix factorization model is defined as Eq. (2), where $\|\cdot\|$ denotes the Frobenius norm. The first three terms in Eq. (2) are used to control the error of matrix factorization. The last term is used to prevent over fitting. The goal of collective matrix factorization is to minimize the objective function, and the algorithm is shown in Fig. 7. It uses gradient descent to get a local optimal solution. The gradients for the four variables (*i.e.*, R, T, F and G) are defined in Eq. (3). In each step, the algorithm firstly searches for the maximal step size (lines 3-6), and then adjusts the four variables according to their gradients (line 7). After the factorization, we can recover $\boldsymbol{X}$ through the production of R and $(T; T)^T$. Then, the recovered entries of $\boldsymbol{X}$ are the estimation results.

$$L(R,T,F,G) = \frac{1}{2}\left\|X - R(T;T)^T\right\|^2 + \frac{\lambda_1}{2}\left\|Y - RF^T\right\|^2 + \frac{\lambda_2}{2}\left\|Z - R(G;G)^T\right\|^2 + \frac{\lambda_3}{2}(\|R\|^2 + \|T\|^2 + \|F\|^2 + \|G\|^2) \tag{2}$$

$$\begin{aligned}
\nabla L(R) &= [R(T;T)^T - X](T;T)^T + \lambda_1[RF^T - Y]F^T + \lambda_2[R(G;G)^T - Z](G;G)^T + \lambda_3 R \\
\nabla L(T) &= [R(T;T)^T - X]^T R + \lambda_3 T \\
\nabla L(F) &= \lambda_1[RF^T - Y]^T R + \lambda_3 F \\
\nabla L(G) &= \lambda_2[R(G;G)^T - Z]^T R + \lambda_3 G
\end{aligned} \tag{3}$$

Algorithm 1: CMF

Input: Incomplete matrix $\boldsymbol{X}$, matrices $\boldsymbol{Y}$ and $\boldsymbol{Z}$, maximum iteration number θ, iteration precision ε

Output: Complete matrix $\boldsymbol{X}$

1: current iteration count $t = 0$

2: **while** $t < \theta$ and $L(R_t, T_t, F_t, G_t) - L(R_{t+1}, T_{t+1}, F_{t+1}, G_{t+1}) > \varepsilon$ **do**

3: initial step size $\gamma = 1$
4: **while** $L(R_t - \gamma\nabla L(R_t), T_t - \gamma\nabla L(T_t), F_t - \gamma\nabla L(F_t), G_t - \gamma\nabla L(G_t)) \geq L(R_t, T_t, F_t, G_t)$ **do**
5: $\gamma = \gamma / 2$
6: **end while**
7: $R_{t+1} = R_t - \gamma\nabla L(R_t)$, $T_{t+1} = T_t - \gamma\nabla L(T_t)$, $F_{t+1} = F_t - \gamma\nabla L(F_t)$, $G_{t+1} = G_t - \gamma\nabla L(G_t)$
8: $t = t + 1$
9: **end while**
10: $\boldsymbol{X} = R_t(T_t; T_t)^T$

Fig. 7. The gradient descent algorithm for collective matrix factorization.

6. EXPERIMENT

6.1 Experimental Setting

6.1.1 Datasets

We evaluated our method based on five real datasets of Hangzhou city, China, which are summarized as follows.

- TCI dataset: It consists of 5305859 TCI records (the sampling interval is 15 minutes) collected from 199 monitored road segments for a period from 16/10/2013 to 03/10/2014. Note that a two-way road segment is treated as two different road segments in the dataset.
- Road network dataset: It is comprised of the 199 monitored road segments, the average length of which is 2.6 km. There are 30 urban express ways, 153 arterial roads and 16 small streets.
- POI dataset: It contains 39305 POIs over 8 categories (*i.e.*, residence, work, commerce, restaurant, school, transportation, entertainment, scenery).
- Check-in dataset: It contains 540395 check-ins collected for a period from 31/10/2013 to 31/10/2014.
- Geotagged photo dataset: It contains 2686 geotagged photos collected for a period from 01/10/2013 to 01/10/2014.

6.1.2 Model configuration

First, as weekdays and weekends usually have different traffic patterns [43], we create matrices for them separately.

Second, there are several parameters in our method, and we configure them as follows. We partition the area of our road network into 16×16 grids. The number of monitored road segments is 199. The number of time slots for TCI data (matrices $\boldsymbol{C}$ and $\boldsymbol{C'}$) is set as 96, as the time span for estimating traffic congestion in the collected dataset is 15 minutes. The number of time slots for mobility features (matrices $\boldsymbol{M}$ and $\boldsymbol{M'}$) is set as 24 (since it results in a relatively higher accuracy).

6.1.3 Evaluation process

The evaluation process is as follows. First, we pick a day from the period of TCI dataset as the estimation day. Then, the data before the estimation day are treated as historical data. The estimation days are chosen from the last months of TCI dataset in the experiments. Second, we use different strategies to generate test cases from the data of the estimation day (*i.e.*, matrix $\boldsymbol{C}$) for the three TCI data estimation problems (as shown in Fig. 8). For data filling problem, we randomly remove α% non-zero entries from $\boldsymbol{C}$. For data inference problem, we completely remove the entries of α% road segments from $\boldsymbol{C}$ (the historical entries of the α% road segments in $\boldsymbol{C'}$ are also removed). For data prediction problem, we completely remove the entries of α% time slots from $\boldsymbol{C}$. Third, our method is used to estimate the values of the removed entries, and the original values of the removed entries are used as ground-truth to evaluate the estimated values. RMSE (*i.e.*, root mean square error) and MAE (*i.e.*, mean absolute error) are used as evaluation metrics (defined in Eqs. (4) and (5)), where n is the number of estimated entries, y_i' is an estimated value and y_i is the ground-truth.

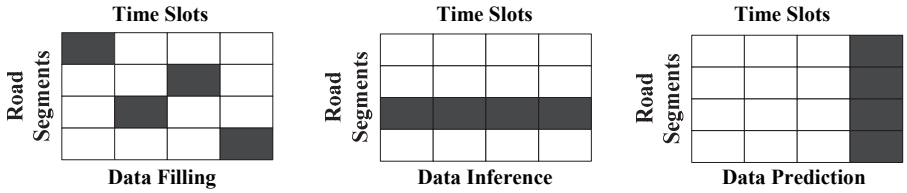


Fig. 8. An illustration of the strategies of test case generation (the gray entries are to be removed).

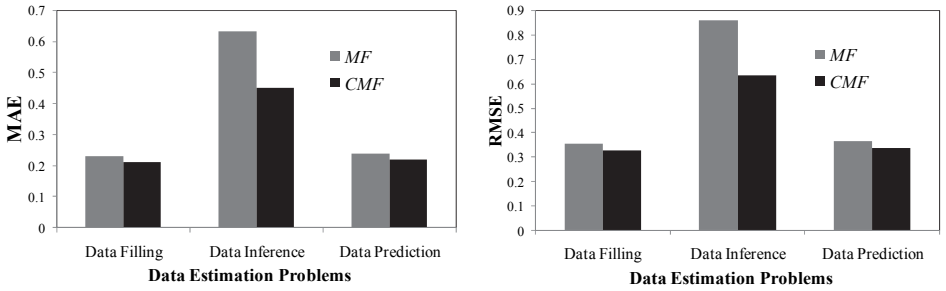
$$MAE = \frac{\sum_{i=1}^{n} |y_i - y_i'|}{n} \tag{4}$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n} (y_i - y_i')^2}{n}} \tag{5}$$

6.2 Evaluation Results

In the first set of experiments, we verify our idea by comparing our method (denoted as *CMF*) with the naïve method that does not take the urban features into account (denoted as *MF*), *i.e.*, only factorizing the TCI data (*i.e.*, matrix $\boldsymbol{X}$). We set $\alpha = 10$, and adopt 10-fold cross validation to generate the test cases. Fig. 9 shows the evaluation results. *CMF* has higher performance than *MF* for all the three TCI data estimation problems. It demonstrates that the urban features are effective in improving the accuracy of TCI data estimation. Moreover, the performance improvement rate of data inference is much higher than that of data filling and data prediction (MAE and RMSE decrease by 8.9% and 8.4% for data filling, 28.6% and 26.3% for data inference, and 7.3% and 7.9% for data prediction). It is because that the historical TCI data of data inference are unavailable for the road segments to be estimated, so it has much more sparse TCI data matrix $\boldsymbol{X}$ than data filling and data prediction. In this situation, the urban features could help

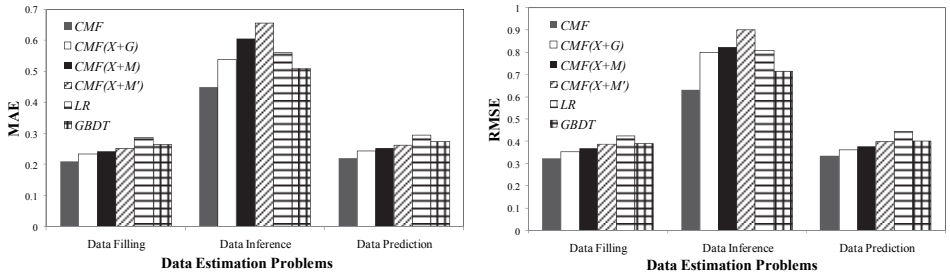
to establish the correlations between road segments to a much greater extent. It also indicates that the historical TCI data has much stronger correlation with the current TCI data than other urban features. The result is consistent with many previous researches [44], which found that the urban traffic conditions usually follow some patterns over a long period of time.



(a) (b)

Fig. 9. The Performance of our method and the naïve method: (a) MAE; (b) RMSE.

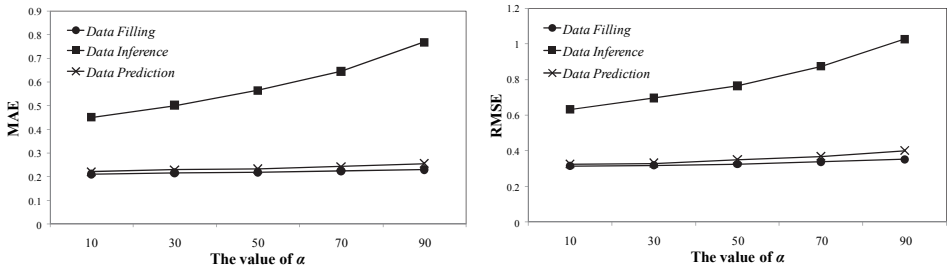
In the second set of experiments, we evaluate the contribution on data estimation of each kind of urban features, *i.e.*, comparing *CMF* with the method that collectively factorizes the TCI data (*i.e.*, matrix $\boldsymbol{X}$) and the geographic features (*i.e.*, matrix $\boldsymbol{G}$), denoted as *CMF*($\boldsymbol{X}$ + $\boldsymbol{G}$), or that factorizes $\boldsymbol{X}$ and the mobility features based on check-in data (*i.e.* matrix $\boldsymbol{M}$), denoted as *CMF*($\boldsymbol{X}$ + $\boldsymbol{M}$), or that factorizes $\boldsymbol{X}$ and the mobility features based on geotagged photo data (*i.e.*, matrix $\boldsymbol{M'}$), denoted as *CMF*($\boldsymbol{X}$ + $\boldsymbol{M'}$). We also compare our collective matrix factorization based model with those regression based models, *i.e.*, *LR* (linear regression) and *GBDT* (gradient boosting decision tree). The regression based models are created as follows. First, we form training instances by concatenating the corresponding values from the TCI data and the urban features. Specifically, a training instance is $\{<t, f_r, f_c, f_p, n_c, n_g>, tci\}$, where t is a time slot, f_r, f_c and f_p are the geographic features of the road segment r, n_c and n_g are the average check-in number and geotagged photo number of r in t, tci is the TCI value of r in t (*i.e.*, the ground-truth). Then, we train regression models from these training instances based on *LR* or *GBDT*. Note that *LR* and *GBDT* use all the urban features for TCI data estimation. The results are shown in Fig. 10. Among the three kinds of urban features, *i.e.*, $\boldsymbol{G}$ (*i.e.*, the geographic features), $\boldsymbol{M}$ (*i.e.*, the mobility features based on check-in data) and $\boldsymbol{M'}$ (*i.e.*, the mobility features based on geotagged photo data), $\boldsymbol{G}$ has the strongest contribution and $\boldsymbol{M'}$ has the weakest contribution. The reason might be that the geotagged photo data usually reflects the mobility of tourists, and thus it is relatively difficult to capture the traffic patterns of local road segments (*i.e.*, road segments outside scenic area). Among the three models, *CMF* (*i.e.*, our collective matrix factorization based model) has the best performance, and *LR* has the worst performance. In addition, model choice has stronger influence on the performance than feature selection for data filling and data prediction. Conversely, feature selection has stronger influence on the performance than model choice for data inference. Since data inference is conducted on sparser TCI data than data filling and data prediction, the results indicate that the effect of these urban features becomes more significant when the available TCI data are of higher sparsity.



(a) (b)

Fig. 10. Comparison of different features and models: (a) MAE; (b) RMSE.

In the third set of experiments, we evaluate the performance of our method by varying the amount of removed entries (*i.e.*, varying the value of α). As shown in Fig. 11, the estimation errors generally increase as more entries are removed (*i.e.*, the value of α increases). However, the increasing trend of data filling and data prediction is fairly stable. It means that our method could well capture the correlations between different dimensions, even the TCI data matrix is highly sparse. On the other hand, α has much stronger influence on the performance of data inference. It might be because that we remove all the TCI data of the road segments. The removed entries are not evenly distributed, but concentrate on some road segments. Consequently, as more road segments were removed, the valid rows of TCI data matrix would become limited, so it would be very difficult to explore the correlations between road segments. On the other hand, the removed entries of data filling and data prediction are scattered over different road segments, so the situation that many road segments are completely "empty" is unlikely to happen, even with a large value of α. Thus, the correlations could still be well established based on the available TCI data and urban features on each road segment.

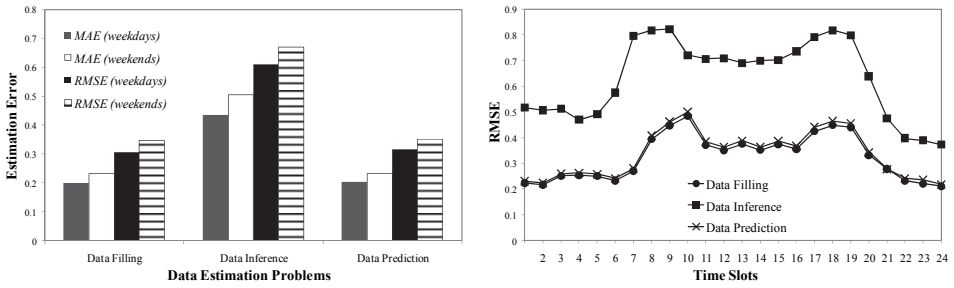


(a) (b)

Fig. 11. The performance by varying the value of α: (a) MAE; (b) RMSE.

In the fourth set of experiments, we explore the performance of our method under different conditions. Fig. 11 (a) shows the performance on weekdays and weekends. It can be found that the performance on weekdays is generally higher than that on weekends. It indicates that the traffic pattern on weekdays is usually more stable than that on weekends. Fig. 11 (b) explores the performance changing over time of day. Since most vehicles do not travel at night, the TCI is highly predictable (*i.e.*, no traffic congestion would happen). Thus, the corresponding RMSE is lower than those at other time slots.

As more vehicles traveling on road surfaces, the traffic patterns on different road segments start to become chaotic. Therefore, the largest RMSEs are observed during rush-hours (*i.e.*, 7am-9am, 5pm-7pm).



(a) (b)

Fig. 12. The performance in different conditions: (a) the performance on weekdays and weekends; (b) performance changing over time of day.

Finally, we evaluate the efficiency of different components of our method, which was implemented based on Java language and MySQL database and tested on a 64-bit PC with a Dual-Core 1.60GHz CPU and 8GB RAM. Table 1 presents the results. First, in these components, only data estimation needs to be executed online (geographic feature extraction and mobility feature extraction could be executed offline). Thus, the execution time is acceptable. Second, the largest proportion of execution time of the offline components (*i.e.*, geographic feature extraction and mobility feature extraction) is cost on querying corresponding objects (*i.e.*, POIs, check-ins or geotagged photos) around a road segment from the database, so it could be reduced by using appropriate spatial index.

Table 1. Efficiency of different components of our method (f_r denotes basic road segment features, f_c denotes spatial closeness features, f_p denotes POI features, M denotes mobility features based on check-ins, M' denotes mobility features based on geotagged photos).

Procedures		Time (ms)	Procedures		Time (ms)
Geographic feature extraction	f_r	269	Mobility feature extraction	M	907974
	f_c	280		M'	12522
	f_p	79904	Data Estimation (*CMF*)		1095

7. CONCLUSIONS

In this paper, we propose a method for estimating the missed TCI data on urban road segments by fusing the sparse TCI data and open data sources (*i.e.*, road network, POIs, check-ins and geotagged photos). By incorporating external correlations from the open data sources, our method could better address the high sparsity problem of TCI data. The experiment results drawn from real datasets have demonstrated that our method has its advantage over the baseline methods when there is a high degree of sparsity of TCI data.

In the future, we will extend our work from the following aspects. First, we will consider more information (*e.g.*, weather, public events, *etc.*) to further enhance the performance. Second, we will exploit the latest studies (*e.g.*, deep learning, granular computing, *etc.*) to upgrade our estimation model.

REFERENCES

1. X. Shan, Z. Wang, and Q. Liu, "Traffic congestion index evaluation based on travel speed on urban expressway," in *Proceedings of the 4th International Conference on Transportation Engineering*, 2013, pp. 1420-1424.
2. Q. Song, M. Li, and X. Li, "Accurate and fast path computation on large urban road networks: A general approach," *PLoS ONE*, Vol. 13, 2018.
3. O. J. Ibarra-Rojas, F. Delgado, R. Giesen, and J. C. Muñoz, "Planning, operation, and control of bus transport systems: A literature review," *Transportation Research Part B: Methodological*, Vol. 77, 2015, pp. 38-75.
4. B. Li, "Recursive estimation of average vehicle time headway using single inductive loop detector data," *Transportation Research Part B: Methodological*, Vol. 46, 2012, pp. 85-99.
5. S. Sivaraman and M. M. Trivedi, "Looking at vehicles on the road: A survey of vision-based vehicle detection, tracking, and behavior analysis," *IEEE Transactions on Intelligent Transportation Systems*, Vol. 14, 2013, pp. 1773-1795.
6. K. Mandal, A. Sen, A. Chakraborty, and S. Roy, "Road traffic congestion monitoring and measurement using active RFID and GSM technology," in *Proceedings of the 14th IEEE International Conference on Intelligent Transportation Systems*, 2011, pp. 1375-1379.
7. H. Xu and J. Ying, "Bus arrival time prediction with real-time and historic data," *Cluster Computing*, Vol. 20, 2017, pp. 3099-3106.
8. X. Kong, Z. Xu, G. Shen, J. Wang, Q. Yang, and B. Zhang, "Urban traffic congestion estimation and prediction based on floating car trajectory data," *Future Generation Computer Systems*, Vol. 61, 2016, pp. 97-107.
9. J. Yuan, Y. Zheng, X. Xie, and G. Sun, "T-Drevie: Enhancing driving directions with taxi drivers' intelligence," *IEEE Transactions on Knowledge and Data Engineering*, Vol. 25, 2013, pp. 220-232.
10. http://www.hzjtydzs.com/.
11. Y. Zhu, Z. Li, H. Zhu, M. Li, and Q. Zhang, "A compressive sensing approach to urban traffic estimation with probe vehicles," *IEEE Transactions on Mobile Computing*, Vol. 12, 2013, pp. 2289-2302.
12. Y. Zhang, M. Roughan, W. Willinger, and L. Qiu, "Spatio-temporal compressive sensing and internet traffic matrices," *ACM SIGCOMM Computer Communication Review*, Vol. 39, 2009, pp. 267-278.
13. G. Adomavicius and A. Tuzhilin, "Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions," *IEEE Transactions on Knowledge and Data Engineering*, Vol. 17, 2005, pp. 734-749.
14. D. Helbing, "Traffic and related self-driven many-particle systems," *Reviews of Modern Physics*, Vol. 73, 2001, pp. 1067-1141.

15. B. D. Greenshields, J. R. Bibbins, W. S. Channing, and H. H. Miller, "A study of traffic capacity," *Highway Research Board Proceedings*, Vol. 14, 1935, pp. 448-477.
16. V. Pattanaik, M. Singh, P. K. Gupta, and S. K. Singh, "Smart real-time traffic congestion estimation and clustering technique for urban vehicular roads," in *Proceedings of IEEE Region Ten Conference*, 2016, pp. 3420-3423.
17. A. Ramazani and H. Vahdat-Nejad, "A new context-aware approach to traffic congestion estimation," in *Proceedings of the 4th International Conference on Computer and Knowledge Engineering*, 2014, pp. 504-508.
18. S. Thajchayapong, W. Pattara-atikom, N. Chadil, and C. Mitrpant, "Enhanced detection of road traffic congestion areas using cell dwell times," in *Proceedings of IEEE Intelligent Transportation Systems Conference*, 2006, pp. 1084-1089.
19. M. Lv, L. Chen, X. Wu, and G. Chen, "A road congestion detection system using undedicated mobile phones," *IEEE Transactions on Intelligent Transportation Systems*, Vol. 16, 2015, pp. 3060-3072.
20. J. Kwon and K. Murphy, "Modeling freeway traffic with coupled HMMs," Technical Report, Department of Computer Science, California University, Berkeley.
21. L. Xu, Y. Yue, and Q. Li, "Identifying urban traffic congestion pattern from historical floating car data," *Procedia-Social and Behavioral Sciences*, Vol. 96, 2013, pp. 2084-2095.
22. B. Pan, Y. Zheng, D. Wilkie, and C. Shahabi, "Crowd sensing of traffic anomalies based on human mobility and social media," in *Proceedings of SIGSPATIAL*, 2013, pp. 344-353.
23. H. Gonzalez, J. Han, X. Li, M. Myslinska, and J.P. Sondag, "Adaptive fastest path computation on a road network: A traffic mining approach," in *Proceedings of VLDB*, 2007, pp. 794-805.
24. P. Wang, T. Hunter, A. M. Bayen, K. Schechtner, and M. C. González, "Understanding road usage patterns in urban area," *Scientific Reports*, Vol. 2, 2012, pp. 1-6.
25. A. Janecek, D. Valerio, K.A. Hummel, F. Ricciato, and H. Hlavacs, "Cellular data meet vehicular traffic theory: Location area updates and cell transitions for travel time estimation," in *Proceedings of UbiComp*, 2012, pp. 361-370.
26. S. Chawla, Y. Zheng, and J. Hu, "Inferring the root cause in road traffic anomalies," in *Proceedings of IEEE 12th International Conference on Data Mining*, 2012, pp. 141-150.
27. Y. Zheng, L. Capra, O. Wolfson, and H. Yang, "Urban computing: Concepts, methodologies, and applications," *ACM Transactions on Intelligent Systems and Technology*, Vol. 5, 2014, pp. 1-55.
28. A. Majid, L. Chen, G. Chen, H. T. Mirza, I. Hussain, and J. Woodward, "A context-aware personalized travel recommendation system based on geotagged social media data mining," *International Journal of Geographic Information Science*, Vol. 27, 2013, pp. 662-684.
29. Y. Zheng, F. Liu, and H. P. Hsieh, "U-air: When urban air quality inference meets big data," in *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2013, pp. 1436-1444.
30. G. Calegari and I. Celino, "Smart urban planning support through web data science on open and enterprise data," in *Proceedings of the 24th International Conference on World Wide Web*, pp. 1407-1412.

31. G. Wilke and E. Portmann, "Granular computing as a basis of human-data interaction: A cognitive cities use case," *Granular Computing*, Vol. 1, 2016, pp. 181-197.
32. Q. Yuan, G. Cong, K. Zhao, Z. Ma, and A. Sun, "Who, where, when, and what: A nonparametric Bayesian approach to context-aware recommendation and search for Twitter users," *ACM Transactions on Information Systems*, Vol. 33, 2015.
33. A. M. Otebolaku and M. T. Andrade, "A context-aware framework for media recommendation on smartphones," in *Proceedings of European Conference on the Use of Modern Information and Communication Technologies*, 2014, pp. 87-108.
34. L. Baltrunas, B. Ludwig, and F. Ricci, "Matrix factorization techniques for context-aware recommendation," in *Proceedings of the 5th ACM Conference on Recommender Systems*, 2011, pp. 301-304.
35. S. Rendle, Z. Gantner, C. Freudenthaler, and L. Schmidt-Thieme, "Fast context-aware recommendations with factorization machines," in *Proceedings of the 34th international ACM SIGIR Conference on Research and Development in Information Retrieval*, 2011, pp. 635-644.
36. D. Yang, D. Zhang, Z. Yu, and Z. Yu, "Fine-grained preference-aware location search leveraging crowdsourced digital footprints from LBSNs," in *Proceedings of ACM International Joint Conference on Pervasive and Ubiquitous Computing*, 2013, pp. 479-488.
37. V. W. Zheng, Y. Zheng, X. Xie, and Q. Yang, "Towards mobile intelligence: Learning from GPS history data for collaborative recommendation," *Artificial Intelligence*, Vol. 184-185, 2012, pp. 17-37.
38. http://www.openstreetmap.org/
39. http://maps.baidu.com/
40. http://weibo.com/
41. http://www.flickr.com/
42. J. Liu, Z. Huang, L. Chen, H. Shen, and Z. Yan, "Discovering areas of interest with geo-tagged images and check-ins," in *Proceedings of the 20th ACM International Conference on Multimedia*, 2012, pp. 589-598.
43. F. Giannotti, M. Nanni, D. Pedreschi, and F. Pinelli, "Trajectory pattern analysis for urban traffic," in *Proceedings of International Workshop on Computational Transportation Science*, 2009, pp. 43-47.
44. C. Peng, X. Jin, K. C. Wong, M. Shi, and P. Liò, "Collective human mobility pattern from taxi trips in urban area," *PLoS One*, Vol. 7, 2012.

Mingqi Lv (吕明琪) received the Ph.D. degree in Computer Science and Technology from Zhejiang University, China. He is currently an Assistant Professor with the College of Computer Science and Technology at Zhejiang University of Technology, China. His research interests include ubiquitous computing and data mining.

Yifan Li (李一帆) received the Bachelor's degree in Software Engineering from Zhejiang University of Technology, China. He is currently pursuing his master's degree in the College of Computer Science and Technology, Zhejiang University of Technology, China. His research interests include data mining and machine learning.

Tieming Chen (陈铁明) received the Ph.D. degree in Software Engineering from Beihang University, China. He is currently a Professor with the College of Computer Science and Technology, Zhejiang University of Technology, China. His research interests include data mining and cyberspace security.

Yinglong Li (李英龙) is currently a Lecturer in College of Computer Science and Technology, Zhejiang University of Technology, China. His research interest is data processing in sensor network.