Inferring Gas Consumption and Pollution Emissions of Vehicles throughout a City

Jingbo Shang[1,2,*], Yu Zheng[2,+], Wenzhu Tong[3,2,*], Eric Chang[2], Yong Yu[1]

[1]Dept. of Computer Science & Engineering, Shanghai Jiao Tong University

[2]Microsoft Research, No.5 Danling Street, Haidian District, Beijing 100080, China

[3]Computer School of Wuhan University, Wuhan, Hubei, China

{v-jishan, yuzheng, v-wetong, echang}@microsoft.com, yyu@apex.sjtu.edu.cn

ABSTRACT

This paper instantly infers the gas consumption and pollution emission of vehicles traveling on a city's road network in a current time slot, using GPS trajectories from a sample of vehicles (e.g., taxicabs). The knowledge can be used to suggest cost-efficient driving routes as well as identifying road segments where gas has been wasted significantly. The instant estimation of the emissions from vehicles can enable pollution alerts and help diagnose the root cause of air pollution in the long run. In our method, we first compute the travel speed of each road segment using the GPS trajectories received recently. As many road segments are not traversed by trajectories (i.e., data sparsity), we propose a Travel Speed Estimation (TSE) model based on a context-aware matrix factorization approach. TSE leverages features learned from other data sources, e.g., map data and historical trajectories, to deal with the data sparsity problem. We then propose a Traffic Volume Inference (TVI) model to infer the number of vehicles passing each road segment per minute. TVI is an unsupervised Bayesian Network that incorporates multiple factors, such as travel speed, weather conditions and geographical features of a road. Given the travel speed and traffic volume of a road segment, gas consumption and emissions can be calculated based on existing environmental theories. We evaluate our method based on extensive experiments using GPS trajectories generated by over 32,000 taxis in Beijing over a period of two months. The results demonstrate the advantages of our method over baselines, validating the contribution of its components and finding interesting discoveries for the benefit of society.

Categories and Subject Descriptors

H.2.8 [Database Management]: Database Applications - data mining, Spatial databases and GIS;

Keywords

Urban computing; traffic pollution; gas consumption; trajectories.

1. INTRODUCTION

While consuming a huge amount of gas, vehicles traveling in cities also generate greenhouse gases (GHG), like CO_2, and air pollution, e.g., PM2.5. Figuring out the answers to challenging questions of emissions, such as how much gas is consumed by vehicles in a given city or on a particular road, or what volume of PM2.5 is generated as

The paper was done when the first and third authors were an intern in Microsoft Research under the supervision of the second author who contributed the main idea and algorithms of this paper.

a direct result, is strategically important to modeling urban traffic, gas consumption, and pollution emission, in order to help save energy and protect the environment [19]. For instance, if we were to know the gas consumption for traveling any road segment at any time, we can suggest cost-efficient driving routes and identify road segments where gas is being wasted significantly. Such knowledge can impact authorities' decision making on improving a city's transportation infrastructure. In the meantime, real-time estimation of emissions from vehicles can enable pollution alerts. In the long run, this information can help diagnose the root cause of air pollution. For example, whether reducing traffic of vehicles will significantly lower air pollution is still a controversial subject in developing countries. There is no clear method to determine the percentage of PM2.5 in a given air space that is generated by vehicles.

As shown in Figure 1, we instantly estimate the total gas consumption and the corresponding pollution emission of vehicles traveling on any road segment in a city at a particular time slot, using the GPS trajectories of a sample of vehicles (e.g. taxicabs). Specifically, we first infer the average travel speed on each road segment, using sparse GPS data from a few vehicles. We then estimate the traffic volume (i.e., the number of vehicles/minute) traversing a road segment based on the average travel speed and other factors, such as weather and type of roads. Finally, the travel speed and traffic volume of a road segment are used to calculate the total gas consumption and vehicle emissions on the road, allowing us to determine the pulse of traffic, energy consumption, and emission levels throughout a city.
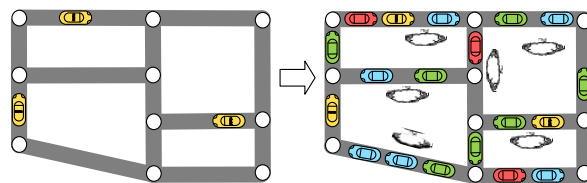

Figure 1. Goals of our Work

Our work faces three main challenges. 1) *Data sparseness*: We can only have a sample of traffic data, as many vehicles do not have a GPS sensor installed. Moreover, in a given time slot, only a small portion of road segments are traversed by GPS-equipped vehicles. For example, according to the statistic on Beijing's taxi data, on average, only 13% of road segments are traversed by taxis in an hour. 2) *From travel speeds to traffic volumes*: The traffic volume on a road segment depends on multiple factors, such as the current travel speeds and density of vehicles, the length, shape, and capacity of the road, as well as weather conditions. Unfortunately, it is not easy to obtain enough training data to learn the mapping between traffic volumes and these factors. In addition, the occurrence of sampled vehicles on roads may be skewed from that of the entire set of vehicles. For example, observing more taxis on a road segment does not mean more occurrences of other vehicles. Consequently, we can neither use the sampled traffic as a training set to learn the

+ *Yu Zheng is the correspondence author of this paper.*

aforementioned mapping, nor simply multiply a factor to the volume of sampled vehicles. 3) *Real-time and citywide*: To enable a valuable service, e.g., enabling air pollution alerts, it is better to have real-time information about traffic, gas consumption and emissions, with a city scale. This calls for a knowledge discovery ability that is both efficient (i.e., providing information on the entire city a few minutes after receiving the data) and effective (e.g., consider the traffic conditions not only on a road segment individually but also on a road network systematically).

Given the aforementioned challenges, existing methods for estimating the traffic conditions on a single road do not work well. For instance, traffic detection solutions [5, 7, 13] using loop sensors on roads and surveillance cameras are difficult to scale up to cover an entire city. Some approaches, e.g., fundamental diagram [1, 3], need a lot of training data, or are not applicable to a complex road network [2]. Conventional models using floating car data to estimate traffic conditions do not tackle the data sparsity problem well, which cannot be simply solved by interpolation [18] or just using historical patterns [2] either (refer to the related work section for details). Our research makes three main contributions:

- We infer the travel speed of a road segment throughout a city, using a model (titled *TSE*) based on a context-aware matrix factorization. *TSE* incorporates historical traffic patterns, the correlation between different times of day, and physical features of a road (such as the length), in a framework of collaborative filtering, to tackle the data sparsity problem.

- We propose an unsupervised graphical model for traffic volume inference (titled *TVI*) on a road segment in a given time slot. *TVI* considers multiple factors, such as physical features of a road, travel speed, variance in speed, and weather conditions, in a Bayesian Network.

- We evaluate the effectiveness and efficiency of our method with extensive experiments, using GPS trajectories generated by 32,670 taxis over a period of two months. Our method outperforms baselines significantly. We have share a sample of the experimental data at [20].

The rest of the paper is organized as follows. Section 2 overviews our method. We detail the methodology in Section 3 and evaluate our approach in Section 4. After reviewing related work in Section 5, we conclude this paper in Section 6.

2. OVERVIEW

Definition 1: Trajectory. A spatial *trajectory Tr* is a sequence of time-ordered spatial points, $Tr: p_1 \rightarrow p_2 \rightarrow \cdots \rightarrow p_n$, where each point has a geospatial coordinate set l and a timestamp t, $p = (l, t)$.

Definition 2: POI. A point of interest POI is a venue (like a school and shopping mall) in the physical world, having a name, address, coordinates, category, and other attributes.

Definition 3: Road Network. A road network RN is comprised of a set of road segments $\{r\}$ connected among each other in a graph format. Each road segment r is a directed edge having two terminal points, a list of intermediate points describing the segment, a length $r.len$, a level $r.lev$ (e.g., a highway or a street), a direction $r.dir$ (e.g., one-way or bi-directional) and the number of lanes $r.n$.

Figure 2 presents the framework of our method, which is comprised of five major components: trajectory mapping, context extraction, travel speed estimation, traffic volume inference, and energy & emission calculations, which will be elaborated respectively in

Section 3. We first map the GPS trajectories received in the current time slot onto a road network using a map-matching algorithm [6], and then calculate the travel speed on the road segments covered by trajectories. We also extract road features from map data, like Points of Interests (POI) and road networks, and traffic patterns from historical trajectories. The road features and traffic patterns are used as context to improve the accuracy of travel speed estimation, which computes the average travel speeds of road segments that are absent of trajectory data through a context-aware matrix-factorization approach. The road features and inferred travel speed as well as other features (like weather conditions) are then employed as observations in a graphical model to infer traffic volumes on a road segment. Finally, employing existing equations from environmental science, we calculate the gas consumption and emission on each road segment.
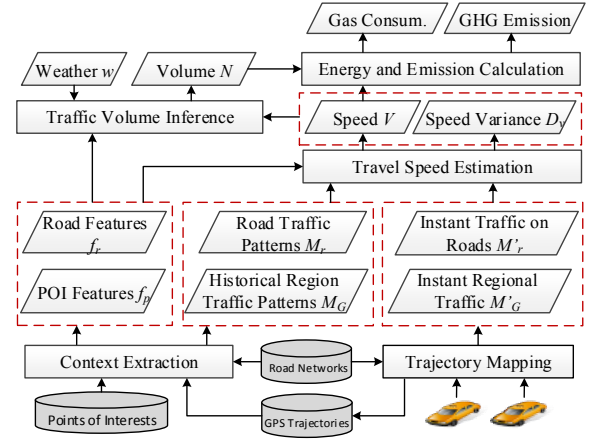

Figure 2. Framework of our method

3. METHODOLOGY

3.1 Trajectory Mapping

The trajectory mapping component receives GPS trajectories from vehicles and projects each trajectory onto a road network using a map-matching algorithm [6]. This component also calculates the average travel speed for road segments currently covered by the trajectory data received. As shown in Figure 3, three vehicles travel four road segments r_1, r_2, r_3 and r_4, generating three trajectories Tr_1, Tr_2, and Tr_3. After map-matching, each point from a trajectory is mapped onto a road segment. Then, we can calculate the travel speed for each point based on Equation 1,

$$v_1 = Dist(p_1.l, p_2.l)/|p_2.t - p_1.t|, \qquad (1)$$

where $Dist$ is a function calculating the road network distance between two points. Likewise, we can calculate the travel speed for other points and then compute the average and variance of travel speed of a road segment as Equation 2 and 3.

$$\bar{v} = \sum_i^n v_i/n; \qquad (2)$$

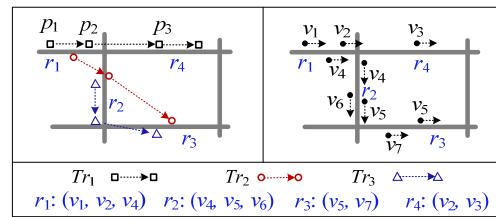$$d_v = \sum_i^n \frac{(v_i - \bar{v})^2}{n}, \qquad (3)$$

Figure 3. Calculating the average travel speed of a road segment

For instance, the average travel speed of r_1 is $(v_1 + v_2 + v_4)/3$. To ensure the quality of the calculated average speed, we require a road segment to be traveled by vehicles for certain times (e.g., 3 times in the experiments). Otherwise, the road segment is considered absent of data and will later be inferred by our method. The average speed and variance formulate a road segment's traffic conditions, which will be used in both travel speed estimation and traffic volume inference components.

3.2 Context Extraction

This component generates two categories of knowledge. One is a set of physical features extracted from POI and road network databases, describing the geographic contexts of a road segment. The other is the traffic patterns learned from historical trajectories, revealing the correlation between different time slots.

Physical features of a road: As shown in Figure 3 A), the physical features of a road segment r further consist of three parts:
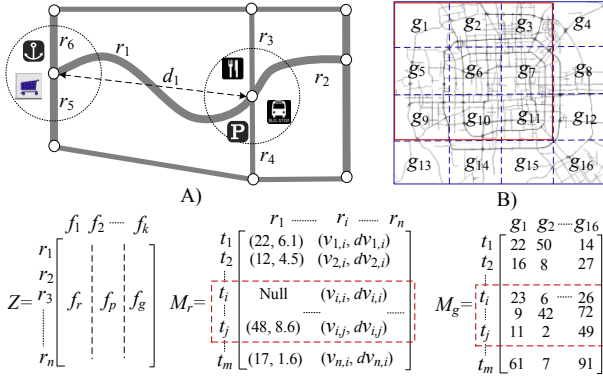

Figure 3. Context Extraction

1) Road network features f_r: $r.len$, $r.lev$, $r.dir$, $r.n$, the number of connections, and a tortuosity τ, which is the ratio between $r.len$ and the Euclidian distance between the two terminals of r. In this example, r_1 has two connections on one terminal and three on the other; $r_1.\tau = r_1.len/d_1$.

2) POI features f_p: For a road segment r, from the POIs around r's terminals (e.g., within a disc of 200 meters), we calculate the distribution of POIs across 10 categories: *Schools*, *Companies & Offices*, *Banks & ATMs*, *Malls & Shopping*, *Restaurants*, *Gas stations & Vehicle services*, *Parking*, *Hotels*, *Residences*, *Transportation*, and *Entertainment & Living Services*. The 10 POI categories are mined from POI and road network datasets, co-occurring with road segments most frequently. For instance, the f_p of r_1 is $(0, 0, 1, 1, 0, 1, 0, 0, 2, 0)$. The feature is further normalized into a distribution.

3) Global position feature f_g: This feature basically indicates which part of the city a road segment falls in. For instance, if we partition a city into 4×4 grids, the f_g of a road segment located in g_6 can be represented by the IDs of its eight neighbors, i.e., $(1,1,1,0,1,0,1,0, 1,1,1,0,0,0,0,0)$, as illustrated in Figure 3 B). If two road segments are geographically close to each other, their f_g will be similar.

We place f_r, f_p, and f_g of each road segment into a matrix Z, where each row denotes a road segment and a column represents a kind of feature. f_r and f_p represent the local features of a road segment, while f_g captures a road segment's global geo-position. The general idea is that road segments with similar f_r and f_p could share a similar traffic condition. If their f_g is also similar (i.e., close to each other), they would have a more similar traffic condition.

Traffic Patterns: While the physical features of a road segment are static, we also extract from historical trajectories two matrices M_r and M_g that respectively represent fine and coarse-grained traffic patterns changing over time of day. Specifically, as shown in the bottom of Figure 3, a row of M_r denotes a time slot (e.g., 2pm-2:10pm), and a column stands for a road segment. Each entry in M_r contains the average traffic condition $(\bar{v}, dv)$ on a particular road segment and in a particular time slot, calculated based on the historical data over a long period (e.g., 2 months). The similarity between two rows reveals the correlation between two time slots. Likewise, matrix M_G reflects coarse-grained traffic patterns in a city, where a row stands for a time slot and a column denotes a grid cell. For example, as shown in Figure 3 B), we partition Beijing into 16 disjoint grids, each of which contains some road segments. Each entry of M_G is the average number of vehicles traversing a specific grid cell in a particular time slot, calculated based on the data over a long period of time. In contrast to M_r, M_g is a higher-level but denser representation of traffic patterns. To model the traffic patterns more accurately, we can maintain M_r and M_g which correspond to workdays and holidays, respectively.

3.3 Travel Speed Estimation (TSE)

This component estimates the traffic condition $(\bar{v}, dv)$ on each road segment, using recently received trajectory data and contexts (extracted in Section 3.2) in a framework of collaborative filtering.

As illustrated in Figure 4, we formulate three matrices X, Y, and Z, where $X = M'_r || M_r$, $Y = M'_G || M_G$, and Z contains the physical features of roads. Specifically, M'_r and M'_G are matrices built based on the recent trajectory data received from t_i to t_j (e.g., 1pm-3pm), where t_j is the current time slot. M_r and M_G are the traffic pattern matrices built over a long period of time, corresponding to the same time slots from t_i to t_j. In the implementation, we build M_r and M_G of an entire day in advance and retrieve the entries from t_i to t_j when constructing X and Y (refer to the broken line box in the bottom of Figure 3). Given the above settings, the goal of estimating current traffic conditions on a road network can be converted into filling the missing values in row t_j of M'_r with the help of M_r, Y and Z.
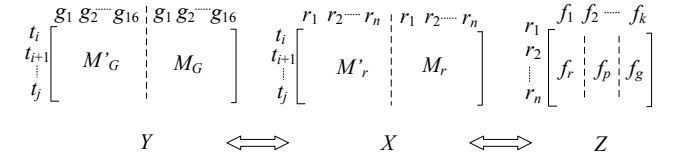

Figure 4. Context-aware Matrix Factorization-based CF

Basically, we can achieve this goal through solely factorizing M'_r into the product of two low-rank matrices based on the non-zero entries of M'_r. However, as we mentioned before, M'_r has very few non-zero entries due to the data sparsity problem. So, the approximation of M'_r by solely factorizing itself is not accurate enough. To tackle the data sparsity problem, we incorporate another two context matrices Y and Z, where Y models the temporal correlations between different time slots and Z models the geographical similarity between two different road segments. Intuitively, M'_r and M'_G represent current traffic conditions in a city, and M_r and M_G denote historical traffic patterns. Putting M'_r together with M_r (and M'_G together with M_G) reveals the deviation of current traffic conditions from the corresponding traffic patterns. Additionally, M_r and M_G built over a long period of time are much denser than the recently received data. As a result, the formulation of X and Y can help tackle the data sparsity problem.

More specifically, we can decompose X, Y and Z as follows:
$$Y \approx T \times (G; G)^T; \ X \approx T \times (R; R)^T; \ Z \approx R \times F^T, \quad (4)$$

where $T, G, R,$ and F are low-rank matrices representing the latent factors; X and Y share latent factor T; X and Z share latent factor R. As Y and Z can be built from other data sources, e.g., historical trajectories and map data, they are much denser than X. Consequently, Y and Z can improve the accuracy of the approximation, if we factorize X, Y and Z collaboratively. After the factorization, we can recover X through the production of T and $(R; R)^T$. The objective function is defined as Equation 5.

$$L(T, R, G, F) = \frac{1}{2}||Y - T(G;G)^T||^2 + \frac{\lambda_1}{2}||X - T(R;R)^T||^2 +$$
$$\frac{\lambda_2}{2}||Z - RF^T||^2 + \frac{\lambda_3}{2}(||T||^2 + ||R||^2 + ||G||^2 + ||F||^2), \quad (5)$$

where $||\cdot||$ denotes the Frobenius norm. The first three terms in the objective function (5) control the loss in matrix factorization, and the last term controls the regularization over the factorized matrices so as to prevent over fitting. In general, the objective function is not jointly convex to all the variables $R, T, G,$ and F. That is, we cannot get closed-form solutions to minimize the objective function. Therefore, we iteratively minimize the objective function according to the gradient descent algorithm shown in Figure 5. Specifically, we have the gradients (denoted as ∇) for each variable:

$$\nabla_T L = [T(G;G)^T - Y](G;G)^T + \lambda_1(T(R;R)^T - X)(R;R)^T + \lambda_3 T,$$
$$\nabla_R L = \lambda_1[T(R;R)^T - X]^T T + \lambda_2(RF^T - Z)F^T + \lambda_3 R,$$
$$\nabla_G L = (T(G;G)^T - Y)^T T + \lambda_3 G, \quad (6)$$
$$\nabla_F L = \lambda_2(RF^T - Z)^T T + \lambda_3 F.$$

Algorithm TSE

Input: Incomplete matrix X, context matrices Y and Z
Output: Complete matrix X.
1. $t = 1$;
2. **While** ($t < N$ and $L_t - L_{t+1} > \epsilon$)   // N is #(max iterations)
3.   Get the gradients $\nabla_{T_t}, \nabla_{R_t}, \nabla_{G_t},$ and ∇_{F_t} by Eq.(6);
4. $\gamma = 1$;
5.   **While**
        $L(T_t - \gamma\nabla_{T_t}, R_t - \gamma\nabla_{R_t}, G_t - \gamma\nabla_{G_t}, F_t - \gamma\nabla_{F_t}) \geq L(T_t, R_t, G_t, F_t)$
6. $\gamma = \gamma/2$;   // search for the maximal step size
7. $T_{t+1} = T_t - \gamma\nabla_{T_t}, R_{t+1} = R_t - \gamma\nabla_{R_t}$ and
8. $G_{t+1} = G_t - \gamma\nabla_{G_t}, F_{t+1} = F_t - \gamma\nabla_{F_t}$;
8. $t = t + 1$;
9. **Return** X;

Figure 5. Model for travel speed estimation

3.4 Traffic Volume Inference (TVI)

3.4.1 Model Description
Intrinsically, there exists a certain relationship, e.g., fundamental diagram, among traffic speed, volume and density. Yet to quantify this relationship accurately requires a large amount of traffic volume data. As many road segments are not equipped with in-road devices, collecting such training data with a city-scale is very costly. Thus, it is impractical to use supervised learning algorithms to learn the relationship. In addition, the occurrence of the sampled vehicles (e.g., taxis) on a road segment may be quite different from the entire set of vehicles, though their travel speed could be similar. In other words, observing more taxis on a road segment does not deduce more occurrences of other vehicles. Given this, we cannot infer the total volume of traffic directly based on the sampled traffic data. To address this issue, we propose an unsupervised graphical model, TVI, based on a partially observed Bayesian Network.

Figure 6 presents the graphical structure of TVI model, where a gray node denotes a hidden variable and white nodes are observations. Note that we can train only one TVI model using the

data of all road segments. The model is then applied to infer the traffic volume for each road segment respectively. Specifically, the traffic volume on each road lane N_a (i.e., the number of vehicles per minute per lane) of a road segment is influenced by four major factors, consisting of the weather conditions w, time of day t, the type of road θ, and the volume of observed sample vehicles N_t. Furthermore, a road's θ is co-determined by its road network features f_r (such as $r.len$), global position feature f_g, and surrounding POIs α which is influenced by f_p and the total number of POIs N_p. $\bar{v}$ and dv are the average travel speed and speed variance, respectively, inferred by TSE model. $\bar{v}$ is determined by θ, N_a, and w. dv is co-determined by N_t, N_a, and $\bar{v}$.
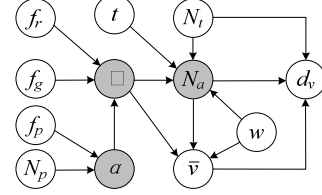

Figure 6. The graphical structure of TVI model

3.4.2 Learning
Due to the hidden nodes, the conditional probability of N_a cannot be drawn simply by counting the occurrence of each condition. Hence, we use the Expectation-Maximization (EM) algorithm to learn the parameters in an unsupervised manner, i.e., without knowing the value of traffic volume N_a, as shown in Figure 7.

Algorithm 2: Parameter Learning of TVI

Input: Bayesian Network structure; observed evidences E
Output: Conditional probability $P(O|Pa(O))$ of each node O
1. Randomly initialize $P(O|Pa(O))$ for each node O
2. **While** $P(O|Pa(O))$ does not converge
3.   **Foreach** evidence $e = (f_r, f_g, f_p, N_p, t, w, N_t, \bar{v}, d_v) \in E$
4.   **Foreach** value of $h = (\alpha, \theta, N_a) \in H$
5. $P(h, e) \leftarrow P(\alpha|f_p, N_p)P(\theta|\alpha, f_g, f_r) \times$
                $P(N_a|\theta, t, w, N_t)P(\bar{v}|N_a, \theta, w)P(d_v|N_a, N_t, \bar{v})$
6.   **Foreach** value of $h = (\alpha, \theta, N_a) \in H$
7. $P(h|e) \leftarrow P(h, e)/\sum_{h \in H} P(h, e)$;
8.   **Foreach** node O
9. $\rho \leftarrow$ the occurrences of $(O, Pa(O))$;
10. $P(O|Pa(O)) \leftarrow \rho/$the occurrence of $Pa(O)$;
11. **Return** $P(O|Pa(O))$

Figure 7. Parameter learning algorithm of TVI

At the beginning part, this algorithm sets the parameters, i.e., the conditional probabilities, e.g. $P(\alpha|f_p, N_p)$ and $P(\bar{v}|N_a, \theta, w)$, with random values. In the E-step (Line 4-7), we call exact inference method, i.e., use the simple Bayesian rule, to compute the values of the hidden nodes (α, θ, N_a) for each instance of the observed data. This is actually an inference process. In the M-step (Line 8-10), by scanning the inferred results from the E-step, the algorithm recalculates the conditional probabilities, which will replace the old parameters. Keep iterating until the parameters converge, we learn a solution for the unknown parameters.

3.4.3 Discretization
All the variables in our model are discretized. This reduces the inference difficulty, while ensuring the inferred results are statistically useful for gas consumption and emission calculation.

Some nodes are discretized based on commonsense knowledge. For example, speed $\bar{v}$ has 7 categories, i.e., [0,10), [10,20), [20,40), [40,60), [60,80), [80,120), [120, ~) km/h, corresponding to different traffic conditions like free flow and congestion. However, it is difficult to find a proper discretization for traffic

volumes (on each lane) N_a, as we do not really have prior knowledge on how big N_a could be on different road segments.

To address this problem, we record 358 video clips on 50 road segments at different times of day, and then manually counted the number of vehicles passing these road segments in a given time slot (i.e. the true traffic volume) (see Table 3 for details) by replaying the video clips. A simple method is to discretize the traffic volume data uniformly according to the range identified from the collected observations. However, the observations are too limited to reveal the true upper bound of the possible volume on road segments. In addition, road segments of different levels may have very different upper bound of traffic volumes. So, we employ the observed true volumes to fit a curve that reveals the distribution of the traffic volumes on different levels of road segments. Then, the fitted distribution is used as a guidance for the discretization of N_a.

As shown in Figure 8, the black data points indicate the observed true volumes, which have been projected onto buckets with a width of 2, i.e., [0, 2), [2, 4), and so on. Significant difference can be observed among the three road levels because of the different road features, such as speed limits and number of lanes. For example, the volume of level 0-1 roads centers in the range of [10, 40), while most volume data of level 3 roads is smaller than 10. To model the traffic volumes on different roads more accurately, we fit a dedicated distributions for each level of road segment (level 0 and 1 share the same distribution as both of them denote highways in Beijing). We adopt normal distribution

$$f(x; \mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} \exp(-\frac{(x-\mu)^2}{2\sigma^2}) \qquad (7)$$

to fit the observed volume data. The parameters are shown in Figure 7, e.g., μ=21.6 and σ=6.38 are good fits for level 0-1 roads (p-value > 0.05 in a chi-square test).
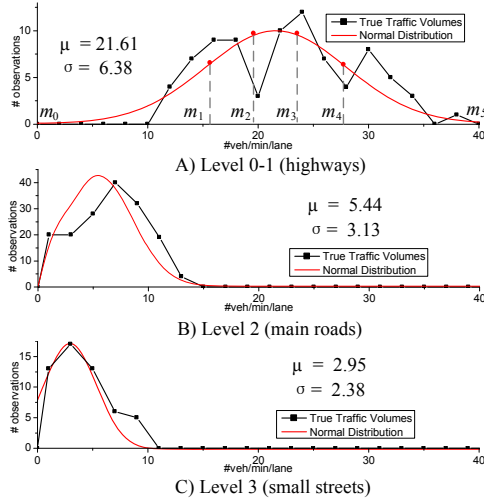
Figure 8. Fitting volume distribution for different road levels

The observations on these road segments can fit the distribution of traffic volume w.r.t. a level of roads for two reasons. First, the selected road segments of a level have a diversity of features (such as length) and traffic volumes. Second, we evenly select the observations from each road segment and in different time slots (82, 163, and 54 observations for level 0-1, 2, and 3 respectively).

Given the fitted distribution of a road level, we find a group of splitting points $m_0, m_1, \cdots, m_5$ that divide the traffic volume into five categories ($c_1, c_2, \ldots, c_5$), such that $\int_{m_{i-1}}^{m_i} f(x; \mu, \sigma) \, dx = 0.2$, where m_{i-1} and m_i are the left and right boundary of category c_i's range respectively; $m_0 = 0$, $m_5 = +\infty$. The five categories may stand for tiny, small, median, big, and very large traffic volumes.

3.4.4 Inference

For a more accurate inference, we train three different models for road segments of level 0-1, level 2, and level 3 respectively, based on the data of the corresponding level. Given a road segment r, we select a model according to $r.lev$ and then infer its traffic volume $r.N_a$ according to the E-step in Line 4-7 of Algorithm 2. The output of the inference is a probability distribution over the 5 discretized volume categories.

We choose $argmax_{c \in \{c_1, \cdots, c_5\}} P(N_a \in c)$ as the inferred category and convert the category c to a real traffic volume according to the corresponding volume distribution $f(x; \mu, \sigma)$ and the inferred probability $P(N_a \in c)$. Specifically, we find the value N_a (in c's range) such that $P(N_a \in c) = \int_m^{N_a} f(x; \mu, \sigma) \, dx / 0.2$, where m is the left boundary of c's range, and 0.2 is the total probability of category c. The total traffic volume on a road segment r equals $r.N_a \times r.n$.

3.5 Energy and Emission Calculation

The information of traffic speed and volume can be further exploited for estimating real-time gas consumption and emissions on road surfaces. Different models are available in environmental science, which quantify the relationship between emissions and speed, as well as other factors, based on large amounts of data. The most frequently used models include MOBILE and COPERT, which have been developed by scientists in the USA and Europe respectively [11]. We use COPERT model [8] because vehicles in Beijing currently adopt European-3 standards [14].

In the COPERT model, traffic emissions consist of three parts, namely hot emissions, cold start emissions, and evaporative emissions. Hot emissions occur when the engine is at its normal mode, which is the general condition for a running vehicle and thus is our major concern. Cold start emissions denote emissions from transient engine operation, and evaporative emissions come from refueling and temperature changes. The latter two parts are omitted in our estimation due to lack of data; they are also of less significance in overall emissions [2].

Hot emission factor (*EF*), the amount of pollutant a single vehicle emits per kilometer (g/km), is calculated as a function of travel speed v (km/h). The generic function is

$$EF = (a + cv + ev^2)/(1 + bv + dv^2). \qquad (8)$$

The parameters are given in Table 1 to calculate different kinds of emissions and gas consumption, which applies for Euro 3 passenger cars (gas consumption calculation additionally requires a vehicle capacity of 1.4-2.0L). Although diversity of vehicles will slightly influence the computation, the results are still statistically useful as we select the most representative cars in the calculation. As for other pollutants like CO_2 and PM2.5, their emission factors are proportional to fuel consumption (*FC*). For instance, the conversion factor of CO_2 is 3.18 for gasoline, i.e., $EF_{CO_2} = 3.18 \times EF_{FC}$; for $PM_{2.5}$ the conversion factor is 3×10^{-5}. The overall emission on a certain road r is:

$$E = EF \times r.N_a \times r.n \times r.len, \qquad (9)$$

Table 1. Pollution emission parameters of COPERT model [8]

	a	b	c	d	e
CO	71.7	35.4	11.4	-0.248	0
Hydrocarbon	5.57×10^{-2}	3.65×10^{-2}	- $1.1\times$	- $1.88\times$	1.25×10^{-5}
Nox	9.29×10^{-2}	- $1.22\times$	- $1.49\times$	3.97×10^{-5}	6.53×10^{-6}
Fuel Consumption	217	9.6×10^{-2}	0.253	- $4.21\times$	9.65×10^{-3}

4. EXPERIMENTS

As the model introduced in Section 3.5 has already been tested in environmental research, we focus on evaluating TSE and TVI.

4.1 Settings

4.1.1 Datasets

Road networks: We use the road network of Beijing, which is comprised of 148,110 nodes and 196,307 edges. The road network covers a 40×50km spatial range, with a total length (of road segments) of 21,895km. Road segments with level 0 and 1 are highways. The bigger the level number is, the smaller the road is.

POIs: The datasets consist of 273,165 POIs of Beijing, which are classified into 195 tier two categories. We only choose the top 10 categories that are located nearby road segments most frequently.

GPS Trajectories: We use GPS trajectories generated by 33,000 Beijing taxis over a period of 47 days. The number of GPS points reaches 673,469,757, and the total length of the trajectories is over 26,218,407km. The average sampling rate is 96 seconds per point.

After projecting the trajectories onto the road network, we come up with the statistics shown in Table 2. "% covered/time slot" denotes the proportion of road segments traveled by at least one taxi in a given time slot (we set 10 minutes a time slot in the following experiments). The last row of Table 2 presents the average number of traverses by taxis in a time slot. Figure 9 A) presents the proportion of road segments traveled by taxis at different times. Even during the peak hours, the proportion is only about 20% on workdays and 17.5% on holidays, respectively. In reality, the travel speed derived from only one taxi's trajectory (i.e., sup=1) is not reliable. If counting the number of road segments traveled by three taxis (i.e., sup=3) in a time slot, the proportion decreases to 8% on workdays and 5% on holidays, respectively. Figure 9 B) further shows the proportion changing over the number of traverses. For example, over 90 percent of road segments are traversed by less than 5 taxis in a time slot. The data presented in Table 2 and Figure 9 reveals the sparsity problem we are facing.

Table 2. Statistics on Beijing road network with mapped trajectories

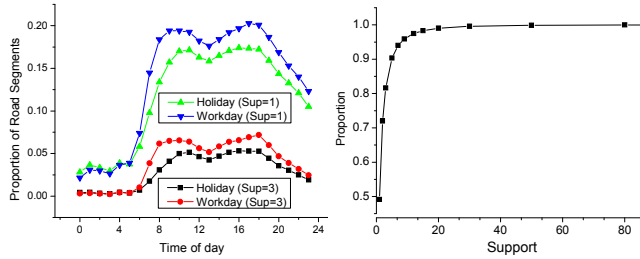|	Level 0,1	Level 2	Level 3	others
Num. of segments	5,061	29,803	142,865	18,578
Total length	1,448km	3,537km	14,747km	2,164km
% covered /time slot	9.97%	7.02%	2.07%	2.94%
# of travels/time slot	1.08	0.42	0.11	0.20

Figure 9. Sparseness of mapped trajectories.

4.1.2 Baselines and Ground Truth

KNN: This baseline uses the average speed and traffic volume of the nearest 3 neighbors to represent a road segment's values.

Kriging is a spatial prediction method with the best linear unbiased estimator [17]. Given a few detected neighbor points $x_i (1 \le i \le n)$, Kriging uses a weighted linear combination estimator to predict an unknown point x_0 as follows: $Z^*(x_0) = \sum_{i=1}^n \lambda_i Z(x_i)$, where $Z(x_i)$ is a value of location x_i, λ_i is a weight, and n is the number of samples. Using a *Kriging* model with a linear kernel, we

interpolate the travel speed of a road segment (absent of taxis) based on its 5 nearest neighbors with data.

Ground Truth: When evaluating TSE, we randomly remove 30% of the non-zero entries from M'_r's last row (i.e., the current time slot) and predict their values using different methods. The 30 percent non-zero entries are then used as a ground truth to measure the accuracy of the predicted values by RMSE (root mean square error), which is defined as

$$RMSE = \sqrt{\frac{\sum_i^n (y_i - \hat{y}_i)^2}{n}} \tag{10}$$

where $\hat{y}_i$ is a prediction and y_i is the ground truth.

The ground truth of TVI is collected as introduced in Section 3.4.3. We calculate the mean absolute error (MAE) of volume on each lane (i.e., number of vehicles per minute per lane) as

$$MAE = |r.N_a - r.\widehat{N_a}| \tag{11}$$

and the mean relative error (MRE) as

$$MRE = \frac{\sum_{r \in RN} |r.N_a - r.\widehat{N_a}| \times r.n}{\sum_{r \in RN} r.N_a \times r.n} \tag{12}$$

where $r.N_a$ is the ground truth of traffic volume on road r's each lane and $r.\widehat{N_a}$ is the prediction; $r.n$ is the number of lanes in r.

Table 3. Ground truth collected for evaluation

Time	7:00 ~ 10:00			10:00~16:00			16:00~20:00			after 20:00			total
Lev.	0,1	2	3	0,1	2	3	0,1	2	3	0,1	2	3	
Holi	0	0	0	6	14	4	6	8	1	4	6	0	49
Work	7	28	8	29	74	9	28	92	7	6	17	4	309
Total	43			136			142			37			358

4.1.3 Configuration of Models

We set 10 minutes as a time slot in our experiment. We partition a city into 4×4 grids and calculate the global position feature f_g and historical traffic pattern M_g (we try different numbers for a partition, finding that 4×4 is slightly better than others). During the experiments, we also found that using 6 time slots (1hour) to formulate M'_r and M'_g results in a relatively higher accuracy. We tried a dynamic extension of our TVI model, and experiment shows that TVI runs much faster without significant accuracy loss. Therefore we keep using the static TVI model in our experiments.

4.2 Evaluation on TSE

In Table 4, we compare the overall performance of TSE with the baseline method that only factorizes current traffic data M'_r, denoted as $MF(M'_r)$, or that combines geographic features Z with M'_r, denoted as $MF(M'_r + Z)$, or that combines the temporal correlation between different time slots of M'_G with Z and M'_r, denoted as $MF(M'_r + M'_g + Z)$. As a result, TSE outperforms all the baselines in terms of RMSE of $\bar{v}$ and dv. The historical traffic patterns that TSE incorporates are very helpful, bringing a significant decrease of RMSE (about 0.5) over $MF(M'_r + M'_g + Z)$. In addition, TSE has very good efficiency, much more so than Kriging. By only using one server (with 4-core, 3.4GHz CPU and 8GB RAM), TSE can infer the travel speed of every single road segment in Beijing within 22 seconds.

Table 4. Overall performance of different methods

Methods	RMSE of $\bar{v}$	RMSE of dv	Time (sec)
$MF(M'_r)$	2.172	1.833	2.2
$MF(M'_r + Z)$	1.939	1.385	18.2
$MF(M'_r + M'_g + Z)$	1.908	1.314	20.2
TSE	**1.369**	**1.035**	**22.2**
Kriging	2.340	1.300	1,000
KNN	3.360	1.590	0.14

Figure 10 A) further explores the performance of TSE changing over time of day. As there are only a few taxis traveling on road surfaces after 10pm and before 8am (see Figure 9 for a proof), the corresponding estimation error is bigger than other time slots. As time goes by, we observe more taxis in the road network (i.e., more road segments are covered by trajectories). Consequently, the RMSE decreases and reaches its best performance at 4pm. While the number of taxis is approaching to its peak at 4pm, the traffic conditions start stepping into congestions. If focusing on estimating the gas consumption and pollution in daytime, our model is more accurate than the results presented in Table 4.
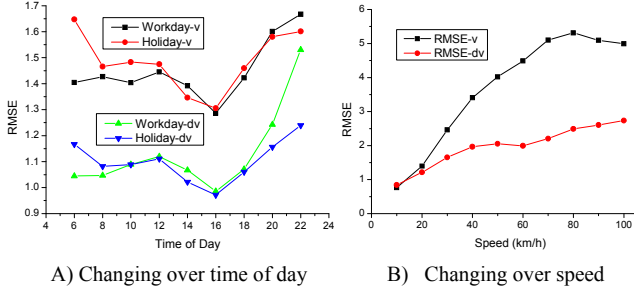

| A) Changing over time of day | B) Changing over speed |

Figure 10. Exploration of TSE's effectiveness

Figure 10 B) shows the RMSE of TSE changing over the (real) travel speed of a road segment to be estimated. Intuitively, a high speed indicates a good traffic condition, in which people have more flexibility in traveling with different speeds. As a result, the variance of speeds (dv) on a road segment can be big and become relatively difficult to estimate. In addition, the number of road segments that allow a high travel speed is small, making it difficult for TSE model to fill in missing values. However, a highway ($\geq$ 90km/h) usually has the lowest driving speed constraint, or people have the concern in mind when traveling on it. This will reduce the flexibility of driving at different speeds, possibly leading to the drop of RMSE after 90km/h in Figure 10 B).

Table 5. Effectiveness of TSE on different levels of road segments

RMSE	level 0-1	level 2	level 3	level >=4
Speed	2.575	1.189	0.977	1.612
Variance	1.526	1.128	0.628	1.234

We further explore the performance of TSE on different levels of road segments in Table 5. It is worth noting that TSE has a good performance on level 2 and 3 road segments, which covers 88.5% of Beijing's road network. These road segments do not usually have a loop sensor. So, it is important to have a method like TSE that can infer the travel speed on such road segments.

4.3 Evaluation on TVI

Table 6 presents the overall performance of TVI. Using the test data presented in Table 3, we compare TVI with six baselines: we first study not using the variance of speed dv (*TVI w/o dv*) and weather conditions w (*TVI w/o w*) in our model, and then compare TVI with several existing methods, e.g., linear regression (*LR*) and fundamental diagram (*FD*).

Clearly, TVI has better performance beyond *TVI* w/o dv and *TVI* w/o w. As traffic conditions depend on the weather condition, e.g., on rainy days the traffic speed is much slower and thus traffic volume is smaller, information on weather conditions does help improve inference accuracy. Besides, having dv together with $\bar{v}$, we can better reveal the traffic conditions on road segments, thereby inferring traffic volumes more accurately. According to the efficiency study, TVI is very efficient, inferring the traffic volume in Beijing's road network in 0.84 seconds, (i.e., $7\mu s$ each).

Table 6. Overall performance of TVI

Methods	MAE	MRE	Inference time (us/road)
TVI	**3.01**	**29%**	**7.27**
TVI w/o dv	3.19	31%	7.18
TVI w/o w	3.15	29%	7.10
LR	3.06	27%	0.15
FD	2.66	16%	0.13
FD-SC	3.9	42%	0.13
FD-DC	6.7	137%	0.13

We also compare TVI with two supervised methods, consisting of linear regression (*LR*) and Fundamental Diagram (*FD*), which learn a curve simulating the relation between the traffic volume and travel speed on a particular road segment based on a large volume of training data. For *LR*, we use half of the data as a training set and another half for a test. W.r.t. *FD*, we select three road segments with enough labels, using half for training and half for testing. Our TVI model outperforms the supervised LR and has a slightly bigger error than FD. Note that TVI is an unsupervised model that can scale up to the entire road network. Designing an unsupervised method with an accuracy even better than a supervised approach is indeed a contribution. In addition, the performance of *TVI* is tested based on the whole dataset presented in Table 1, while *FD* is only tested on three roads and *LR* is on half roads. Tested on the same three road segments, TVI has a volume error of 2.73 which is even closer to that of *FD*. It should be noted that *FD* needs training for each individual road segment, some of which may have no volume data at all. To address this issue, we first cluster road segments into a few groups in terms of their geographical features in matrix Z. We train a *FD* model based on the data from some road segments in a cluster, and then apply the model to other road segments in the same cluster (*FD-SC*). The performance decreases very quickly. If applying the model trained from one cluster to another cluster, the volume error is even bigger. These results denote that *FD* is not scalable to an entire road network.

Table 7 further explores the performance of TVI on different road levels and in different type of days. It turns out that TVI has a better performance on level 2 road segments which are absent of a loop sensor but with a large volume of traffic, and thus are our major concerns. Additionally, TVI has a better performance on weekdays than weekends, as some taxi drivers do not work on weekends, and traffic conditions on weekends are relatively irregular and complex compared to weekdays.

Table 7. Exploration of the performance of TVI

	level 0-1	level 2	Weekday	Weekend
MAE	5.55	2.23	2.97	3.28
MRE	22%	41%	29%	30%

Table 8 further presents the time that different components spend on processing the data of a time slot (i.e., 10 minutes), if we only use a single core of a single server. Note that we can map-match the trajectories of different drivers separately and infer the traffic volume on different road segments independently. Using five cores of the server to perform the map-matching and TVI in parallel, we can infer the travel speed, traffic volume, energy consumption, and pollution emission throughout Beijing's road network in 1 minute.

Table 8. Efficiency of different components of our method

Online components	Time	Offline components	Time
Map-matching	4.94min	Geo-feature extraction	149s
TSE	22.2s	Historical pattern extraction	240s
TVI (inference)	0.84s	TVI learning	89s
Total	5.32min	Total	478s

4.4 Visualization

Figure 11 visualizes the traffic volume, energy consumption, and pollution emission around Beijing Olympic Park on a weekday (2013/9/17 Tuesday), weekend (2013/9/21 Saturday), and public holiday (2013/10/2), respectively.
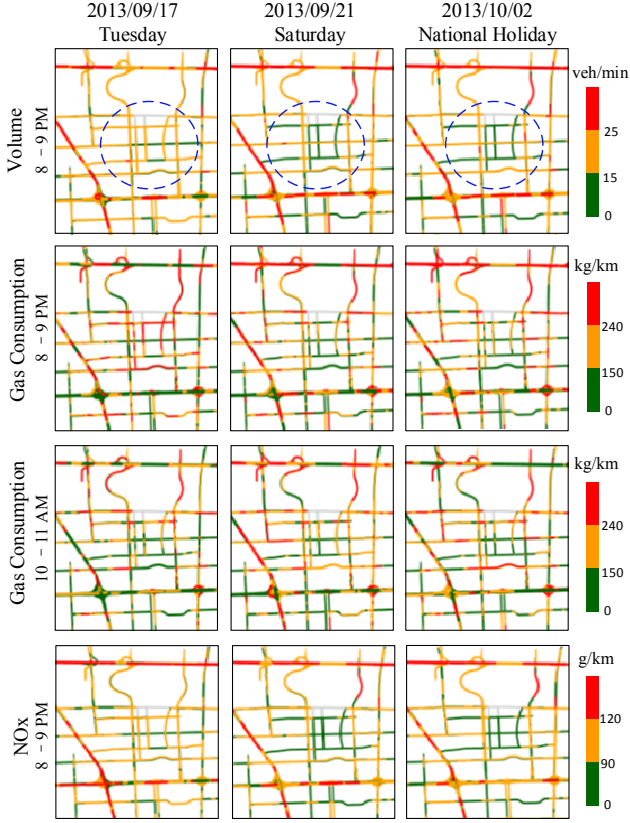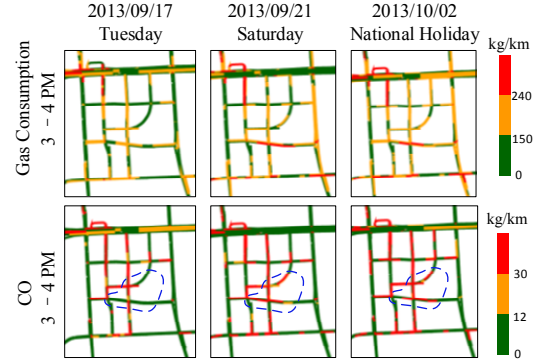

Figure 11. Traffic volume, energy consumption and emissions around Beijing Olympic Park in different time slots

As depicted in the first row, in time slot 8-9pm, the region has a bigger traffic volume during the weekday than during the weekend and holiday. This is aligned with our commonsense knowledge that people usually visit this region in the daytime (rather than in the evening) of weekends or a holiday. The large volume of traffic on a weekday is caused by people who drive home after work passing by this region. Given the same reason, in the second row of Figure 11, we observe more red road segments, which denotes more energy consumption, on the weekday than on the weekend and the national holiday. The latter two days generally have a similar energy consumption pattern except several road segments where the national holiday is slightly higher than the weekend. The former has relatively more sightseeing buses arriving at the park or more tourists taking taxis to this region in the evening (than the latter). On the contrary, in the time slot from 10 to 11am, we observe more energy consumption on the weekend and holiday, as many people drive to the park at these times for fun. On a workday, after morning rush hours (e.g., 8-10am), people barely drive to this place during working time. The emissions of $PM_{2.5}$ and CO_2 have the same pattern as the energy consumption, just having different factors (3×10^{-5} and 3.18 respectively). The last row of Figure 11 presents the emission of NO_x in the region, where the workday has more emission than the weekend and national holiday in 8-9pm.

Figure 12 demonstrates the gas consumption and emission of NO_x around Zhongguancun area, which is a place mixed with many companies and entertainments, in the aforementioned three days,

respectively. In the time slot from 3pm to 4pm, the time before evening rush hours, this area has less gas consumption on the workday than during the weekend and holiday, because people are still working indoors. When time goes to weekends and holidays, many people travel to this region for the purpose of entertainment, e.g., go shopping and watch a movie, leading to more energy consumption and emission of CO, as illustrated in the two rows of Figure 12. There is a movie theater, a supermarket, and two shopping centers located in the region marked by the broken curve.

Figure 12. Gas consumption and pollution emission around Zhonguancun area

Figures 13, 14 and 15 show the citywide traffic volume, gas consumption, and NO_x emission on three types of days: weekdays, weekends, and public holidays. The number plotted in the three figures are averaged by day, respectively. Figure 13 presents the average traffic volume on a road segment in one minute, which depends on the travel speed on the road segment and the density of the traffic. Before 7am, a workday has a larger volume of traffic than a weekend or a holiday. At this moment, the travel speed is very fast during the three types of days, while more people travel (to work) on a weekday than a holiday. When time goes to 8-10am (i.e., Beijing's morning rush hours on workdays), however, the travel speed decreases significantly on a weekday, reducing the traffic volume that can pass a road segment. The weekend and holiday have a similar decreasing trend, but do not drop tremendously, because the travel speed is not that slow from 8am to 10am. After morning rush hours, the travel speed increases, leading to an increase of the traffic volume. Then, we see the decreasing pattern repeats in evening rush hours, 5-7pm, during a weekday. After evening rush hours, the traffic volume increases again with the increase of the travel speed on a weekday, until at later night when the traffic density decreases tremendously (we do not show the traffic volume after 11pm in these figures as they are very small and quite similar to each other). The public holiday has a bigger traffic flow than the other two days, as a portion of people leave Beijing for vacation, which reduces the traffic density slightly but enhances the travel speeds on roads significantly.

Figure 14 shows the total energy consumption of the entire road network per hour changing over time of day. For instance, from 12pm to 1pm, about 3,850,000 liters of gasoline are consumed by vehicles in Beijing. Supposing there are 2 million vehicles traveling on the road network in this time slot, each vehicle consumes 1.9 liter gasoline per hour, which is about 19 kilometers per hour. A peak occurs in this time slot due to the bigger traffic volume and travel speed. The reason a public holiday consumes more energy may lie in the fast travel speed and a farther distance that people would travel during holidays.

Figure 15 presents the total emission of NO_x in Beijing's road network, where weekends have the largest volume in the afternoon.
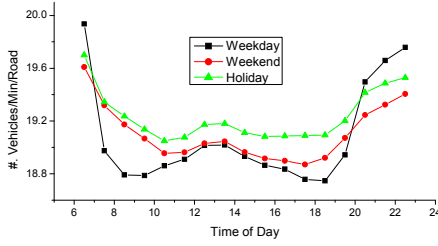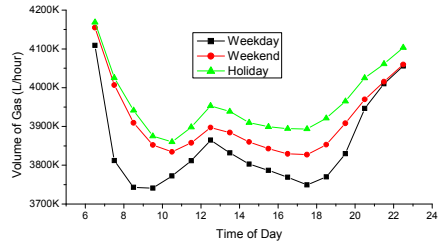
Figure 13. The citywide traffic volume of Beijing
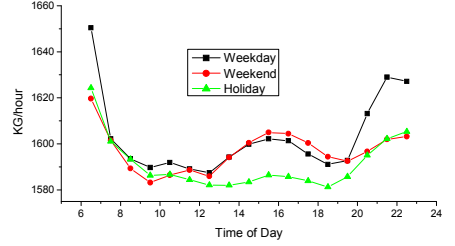

Figure 14. Citywide gas consumption
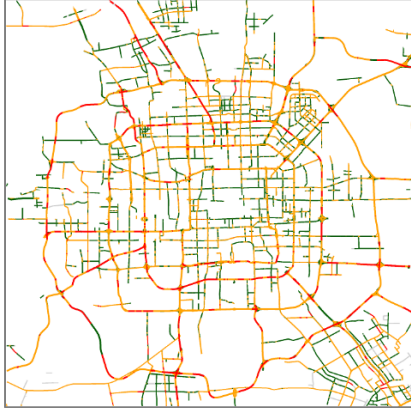

Figure 15. Citywide NOx emission

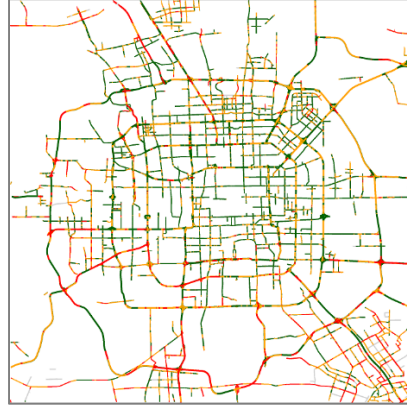**Figure 16. Geo-distribution of Traffic Volume**

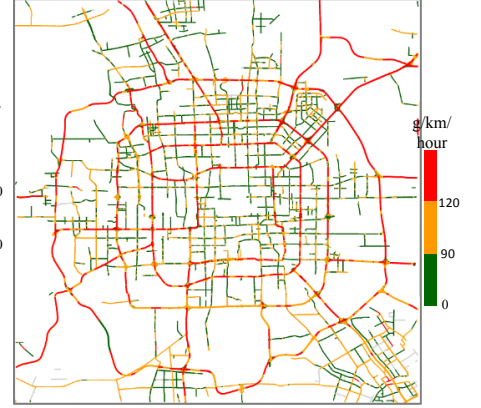**Figure 17. Geo-distribution of energy consumption**

Figure 18. Geo-distribution of NOx emission

Beijing is running a traffic control policy based on the license ID of a car on workdays. As this policy does not apply to weekends and holidays, the number of people who travel by car is even bigger on a weekend. On public holidays, many people leave the city for vacation, therefore the emissions are not heavy. Combining Figure 14 and 15, we find that public holidays have more gas consumption but less emission than weekdays and weekends, i.e., the energy has been used in a cleaner way.

Figure 16 visualizes the mean traffic volume (i.e., the average number of vehicles per minute) traversing every road segment in Beijing from 6am to 10pm on workdays. The road segments with relatively large volumes of traffic (marked red) are mainly located in the south (e.g., the southern segments of the fifth ring road) and some highways spreading towards rural areas. As the major business and entertainment areas are located in the northern and central part of Beijing, the average travel speed in these places are generally slower than the southern part. The slow speed reduces the traffic volume on corresponding road segments. The traffic volume on the northern segment of the fourth ring road is relatively larger than other road segments in the north part, which well matches our commonsense knowledge. Figure 17 presents the average energy consumption of vehicles on each road segment per hour, from 6am-10pm. As the total energy consumption depends on both travel speed and traffic volume, some road segments marked red in Figure 16 become green in Figure 17. Given this kind of visualization, urban planners and transportation authorities can identify the road segments that have wasted unnecessary energy (e.g., a road segment with small traffic volume while having a large energy consumption), therefore informing future urban planning. Figure 18 displays the total NO_x emission on road segments per hour in the same time slot, where the ring roads and highways have more emissions. When the emission on some road segments exceeds a threshold, we can send alerts to people passing by or living around. The information can also guide drivers for finding the best route in terms of the energy consumption, air quality, and travel time.

5. RELATED WORK

5.1 Traffic Condition Modeling

Traffic modeling on individual roads: Conventional methods use traffic monitoring cameras (usually combined with a speedometer) to observe speed and volume data where relevant instrumentation is installed. This kind of approach usually employs a Fundamental Diagram [1, 3] to learn a specific relation among travel speed, traffic density, and volume for a particular road, from a large amount of data collected by the camera and speedometer. As the number of vehicles traversing a road may not be perfectly detected by using automatic computer vision algorithms, in most cases, human effort is needed to count the vehicles so as to generate training data for the fundamental diagram. Consequently, this kind of methods is difficult to scale up to an entire city, let alone the limited coverage of traffic monitoring camera and speedometers. Later, in-road loop sensors are widely deployed to detect both travel speed and traffic volume. The rich data enables sophisticated modeling to simulate traffic status between the instrument [7, 13], or predict future traffic status [5], for a specific road where the device is installed. Unfortunately, the majority of these models do not incorporate the correlation between the traffic conditions of different road segments (especially those that are geospatially disconnected), when estimating the traffic on a road segment. Some methods [5] model several connected road segments simultaneously with a Markov Model. However, applying these methods to the entire road network will result in a huge model that has very poor efficiency and big issues with training parameters. Additionally, the coverage of loop-sensor systems is also limited due to the high expense for device installation and maintenance.

Traffic modeling on a road network: Recently, more traffic modeling systems have turned to using Floating Car Data [10, 15], which is generated by vehicles traveling around a city with a GPS sensor. The trajectories of these vehicles will be sent to a central system and matched onto the road network for deriving speed on road segments. However, due to the uneven distribution of the

probing vehicles as well as the low sampling rate of GPS data (e.g., one point per minute), many road segments are not covered by any trajectory [17]. To address this problem, some interpolation-based methods, such as KNN and Kriging [18], have been proposed to infer the traffic conditions of a road segment based on the values of its spatial neighbors. More advanced methods model the traffic conditions on a road network with a road-time matrix, where each entry stands for the traffic conditions at a specific road segment and in a particular time slot. These methods then adopt Compressive Sensing-based algorithms [16, 17] to fill in the missing entries in the matrix. The interpolation-based methods consider the spatial correlation among different locations' traffic conditions, and the compressive sensing-based approach considers both spatial and temporal correlations. Unlike these methods, our model incorporates two additional (but very helpful) correlations from extra data sources. One is the correlation between road segments in terms of their geographic features, such as the length, POI distributions, and number of connections, learned from road network and POI datasets. The geographic features capture the local and global similarity between two road segments simultaneously. The other is the correlation between the current and historical traffic conditions, learned from historical trajectories over a period of time. Extensive experiments validate the contribution of these correlations beyond the original correlation solely learned from the road-time matrix.

5.2 Energy and Emission Estimation

Vehicular fuel consumption and pollutant emission is a critical environmental issue nowadays. [9] studies the empirical relation between traffic emissions and travel speed as well as the acceleration for a single vehicle. [2] exploited a framework similar to ours to estimate macroscopic pollutant emissions. The main concern with this method is the use of the historical average to deal with the aforementioned data sparsity problem. Consequently, irregular conditions (e.g., congestions caused by accidents) are not well captured. In addition, it applies a homogeneous FD to different roads in the same cluster, which has been compared with our method as a baseline (*FD-SC*).

There also exist crowdsensing-based approaches [4, 12] that sense environmental data using people or vehicles carrying air quality sensors. These methods cannot bypass the data sparsity problem either when scaling up to a citywide application, as the number of sensors is even smaller than the number of GPS-equipped vehicles. These approaches also do not answer the questions, such as how much pollutant is generated from traffic.

6. CONCLUSION

In this paper, we instantly infer the traffic volume, energy consumption, and the emissions of vehicles traveling on a city's road networks, based on the GPS trajectories received from a sample of vehicles, e.g., GPS-equipped taxicabs, in current time slot and over a period of history. The knowledge derived from our research can enable many valuable applications for social good, such as timely road-level pollution alerts, monitoring citywide traffic conditions, improving urban planning, and helping study the root cause of air pollution. This is a very challenging problem, however, considering its efficiency, effectiveness, and scalability. We address this problem through a three step approach, where we first propose TSE model to compute the travel speed on each road segment based on incomplete trajectory data, then infer the volume of a road segment by using TVI model, and finally calculate the energy consumption and pollution emission based on the inferred travel speed and traffic volume. We evaluate our approach based on extensive experiments that use GPS

trajectories generated by over 32,000 taxicabs over a period of two months. The results demonstrate the effectiveness, efficiency and scalability of our method, which outperforms baseline approaches, such as Linear Regression, Fundamental Diagram, and Kriging. Using five cores of a server, we can grasp the traffic conditions, energy consumption, and emissions of the past 10 minutes throughout Beijing's road network within 1 minute.

REFERENCES

[1] Greenshields, B. D., Bibbins, J. R., Channing, W. S., and Miller, H. H. 1935. A study of traffic capacity. In Highway research board proceedings (Vol. 14).

[2] Gühnemann, A., Schäfer, R. P., Thiessenhusen, K. U., and Wagner, P. 2004. Monitoring traffic and emissions by floating car data. Institute of Transport Studies Working Paper, (ITS-WP-04-07).

[3] Helbing, D. 2001. Traffic and related self-driven many-particle systems. Reviews of modern physics, 73(4), 1067.

[4] Honicky, R., Brewer, E. A., Paulos, E., and White, R. 2008. N-smarts: networked suite of mobile atmospheric real-time sensors. In Proc. of the second ACM SIGCOMM workshop on Networked systems for developing regions (pp. 25-30).

[5] Kwon, J., and Murphy, K. 2000. Modeling freeway traffic with coupled HMMs. Technical report, Univ. California, Berkeley.

[6] Lou, Y., Zhang, C., Zheng, Y., Xie, X., Wang, W., and Huang, Y. 2009. Map-matching for low-sampling-rate GPS trajectories. In Proc. of ACM SIGSPATIAL GIS, 352-361.

[7] Muñoz, L., Sun, X., Horowitz, R., and Alvarez, L. 2003. Traffic density estimation with the cell transmission model. In Proc. of the 2003 American Control Conference, Vol. 5, 3750-3755.

[8] Ntziachristos, L., Samaras, Z., Eggleston, S., Gorissen, N., Hassel, D., and Hickman, A. J. 2000. COPERT III. Computer Programme to calculate emissions from road transport, methodology and emission factors (version 2.1). European Energy Agency (EEA), Copenhagen.

[9] Panis, L.I., Broekx, S., and Liu, R. 2006. Modelling instantaneous traffic emission and the influence of traffic speed limits. Science of the total environment, 371(1), 270-285.

[10] Schäfer, R. P., Thiessenhusen, K. U., and Wagner, P. 2002. A traffic information system by means of real-time floating-car data. In ITS world congress (Vol. 2).

[11] Smit, R., Ntziachristos, L., and Boulter, P. 2010. Validation of road vehicle and traffic emission models–A review and meta-analysis. Atmospheric environment, 44(25), 2943-2953.

[12] Steed, A., Spinello, S., Croxford, B., and Greenhalgh, C. 2003. E-Science in the streets: urban pollution monitoring. In UK e-science all hands meeting.

[13] Wilkie, D., Sewall, J., and Lin, M. 2013. Flow reconstruction for data-driven traffic animation. ACM Trans. on Graphics, 32(4), 89.

[14] Xie S., Song X., and Shen X. 2006. Calculating Vehicular Emission Factors with COPERTⅢ Mode in China. Environmental Science, 27(3), 415-419.

[15] Yuan, J., Zheng, Y., Zhang, C., Xie, W., Xie, X., Sun, G., and Huang, Y. 2010. T-drive: driving directions based on taxi trajectories. In Proc. of the 18th SIGSPATIAL GIS, 99-108.

[16] Zhang, Y., Roughan, M., Willinger, W., and Qiu, L. 2009. Spatio-temporal compressive sensing and internet traffic matrices. In ACM SIGCOMM Computer Communication Review, 39(4), 267-278.

[17] Zhu, Y., Li, Z., Zhu, H., Li, M., and Zhang, Q. 2013. A compressive sensing approach to urban traffic estimation with probe vehicles. Mobile Computing, IEEE Transactions on, 12(11), 2289-2302.

[18] Zou, H. X., Yue, Y., Li, Q. Q., and Yeh, A. G. O. 2011. Traffic data interpolation method of non-detection road link based on Kriging interpolation. Jiaotong Yunshu Gongcheng Xuebao, 11(3), 118-126.

[19] Zheng, Y., Capra, Li, Wolfson, O., Yang, H. 2014. Urban computing: concepts, methodologies, and applications. ACM Trans. On Intelligent systems and Technology, 5(3).

[20] Data released: http://research.microsoft.com/apps/pubs/?id=217455