Traffic Flow Estimation using Probe Vehicle Data

Olga Gkountouna
George Mason University
Fairfax VA, USA
ogkounto@gmu.edu

Dieter Pfoser
George Mason University
Fairfax VA, USA
dpfoser@gmu.edu

Andreas Züfle
George Mason University
Fairfax VA, USA
azufle@gmu.edu

***Abstract*—Traffic sensing has been revolutionized with the commoditization of GPS technology. Smartphone navigation applications ubiquitously track vehicles as samples of the overall traffic. This so-called Probe Vehicle Data (PVD) has replaced traditional road-side sensor technologies, such as induction loops and microwave sensors, given its relative low cost, good coverage, and reliability. However, while PVD allows us to assess speed and by extension the overall traffic condition in a road network, this sample-based approach does *not* provide us with traffic flow, i.e., the number of vehicles passing through an edge of the road network. This paper bridges this gap by proposing and evaluating a range of methods to infer traffic flow for a road network that is ubiquitously observed using probe data but having traffic flow measurements only in very road-side sensor locations. We create Road Segment Archetypes that relate PVD speeds to flow from road-side sensors for these locations. These archetypes are then extended to the entire network covered only by PVD based on similar traffic characteristics. Using these archetypes we augment and experimentally evaluate different traffic flow estimation models using real-world traffic data. Experimental results show that the Road Archetype flow estimation is comparable to the accuracy of prediction models that would be based on actual road-side sensor flows.**

***Index Terms*—Modeling, Estimation, Traffic Flow, Road Networks, Road Segment Archetypes, Probe Vehicle Data**

I. Introduction

With the proliferation of smartphones, Probe Vehicle Data (PVD) has become the prevalent means for monitoring traffic conditions. Such probe data provides speed information for the entire road network. However, it does not provide actual traffic flow as the fraction of vehicles in traffic captured by PVD varies considerably over space and time as shown in [44] and it is not trivial to estimate the actual number of vehicles in traffic. Traditional means for assessing traffic have been stationary roadside sensors such as induction loops and microwave sensors, which capture accurate traffic flow data at discrete locations throughout the road network and carefully chosen by traffic management authorities. Due to the associated cost and unreliability of such a sensor network, it is infeasible to have complete coverage, and thus, traffic models are used to infer and estimate flows in networks based on such sensor readings. Yet, understanding the traffic flow throughout a network would benefit transportation planning [4] to efficiently adapt the transportation infrastructure to existing needs. In addition, having accurate estimates of traffic flow has applications in traffic [26] and emissions prediction [42].

In this work, our goal is to combine PVD with sparse stationary roadside sensors and to build a model that estimates (rather than measures) *traffic flow in the entire network* based on available PVD. To train models to infer traffic flow from traffic speed, we leverage a small number of stationary roadside sensors to measure flow. To estimate traffic flow in other locations, we enrich speed information with learned periodic (daily, weekly, etc.) patterns of human mobility, as it has been shown that "daily mobility is, in fact, characterized by a deep-rooted regularity" [17], [45]. We assess various statistical and machine learning models with respect to their suitability to estimate traffic flow based on a learned speed-flow relationships.

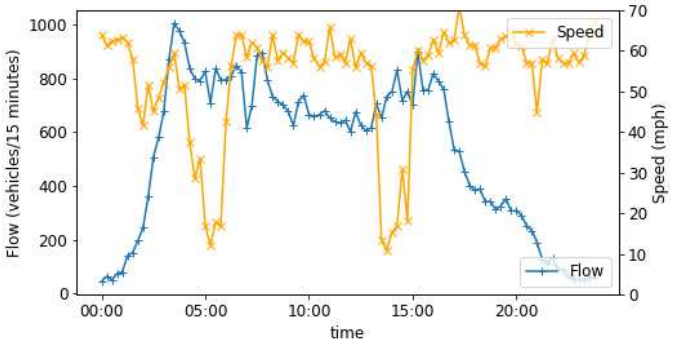


Fig. 1. Traffic Flow and Traffic Speed on Interstate Road I-395 heading north, January 24th 2017.

As speed information is available almost for the entire network, infer traffic flow from traffic speed and a given road capacity is not a trivial problem. Consider the a $15min$ interval of speed and flow data for a typical Tuesday on an interstate route near Washington D.C. shown in Figure 1. We observe two major drops in speed; early morning and late afternoon periods, which coincide with the peak traffic conditions. During such traffic jams, the speed can drop to as low as $10mph$. Yet, we do not observe a drop in traffic flow. While vehicles move many times slower during these periods of congestion, they also move much closer to each other, thus leading to a higher traffic density, effectively compensating for the effects of decreased speed. A takeaway here is that speed does not always correlate with flow.

Traditional approaches used to capture the speed-flow relationship, such as the Bureau of Public Roads (BPR) function [7], are prone to considerable errors. Such approaches model this relationship based on the road characteristics and categories. Specific parameters such as the coefficient and the

exponent of flow in these equations are recorded in relation to road categories (highway, freeway, etc.). However, even after tuning these parameters using historic measurements of speed and flow for specific roads, large errors are still possible in the estimation of traffic flow.

In this work, we use ubiquitous PVD speed information for the entire road network in conjunction with a limited number of flow measurements from stationary sensors throughout the network. Using additional information such as time and road information, we test several machine learning approaches to *extrapolate flow information for the entire network*. We devise so-called road segment archetypes that signify speed-flow relationships for prototypical locations, and use those to train models in an effort to estimate the flow for arbitrary locations in the road network. The primary goal of this work is to propose a system to estimate traffic flow ubiquitously from sparse traffic flow measurements and ubiquitous speed measurements from PVD. Our experiments for the Washington D.C. metropolitan area compare and evaluate different flow estimation models, and augment these models using our segment archetype approach. To evaluate our method, we use traffic flow data provided by the Virginia Department of Transportation (VDOT) for the greater Washington D.C. metro area and PVD data provided by INRIX. Our experiments show that our proposed approach drastically outperforms traditional solutions to estimate traffic flow from traffic speed, showing a proof-of-concept that our proposed system can be applied to other cities where road-side sensors are sparsely placed to train our proposed segment archetype approach.

To summarize, our contributions are:

- based on vehicular speed and flow, we categorize road segments into archetypes;
- we model traffic flow of these archetypes based on vehicular speed and flow periodic patterns;
- using these models, we experimentally provide a comprehensive perspective on the strengths and weaknesses of state-of-the-art prediction and regression models;
- our evaluation uses actual traffic flow and speed measurements for the Northern Virginia/Washington D.C. metro area.

The remainder of this paper is organized as follows. After a survey of related work in Section II, we formalize our problem in Section III. Section IV describes our latent feature extraction and clustering approach to model different groups of sensor locations. Finally, in Section V, we show the results of our experimental evaluation, comparing different regression and prediction algorithms to estimate flow from traffic speed.

II. Related Work

Our goal is to estimate traffic flow from publicly available data sources. Towards this goal, the fundamental diagram [18], [22], [30], [32] describes the relationship between traffic density, speed and flow. It can be used to infer vehicular flow from traffic speed and density. The Bureau of Public Roads (BPR) developed a link congestion function, the BPR curve [7], which describes traffic speed as a power law of the flow to capacity ratio. It includes parameters related to the type of road, as well as the capacity and free-flow speed. Such functional approaches, while applicable ubiquitously, introduce large errors, as they cannot learn from available flow information given by sparsely placed road-side sensors. Our experimental evaluation in Section V supports this claim by showing that even in the case where the parameter of these functions are guessed optimally, the prediction results of BPR function-based solutions are far inferior to our proposed data-driven solutions.

Traffic flow, congestion, and other aspects of transportation networks have been traditionally measured using static road-side sensors [23], [32], [36], [50] and surveillance cameras [3], [46], [53]. The respective flow measurements can be used to train models in order to predict future traffic flow [24], [35], [50]. However, these quantities are not readily available for the entire road network given that deploying and maintaining a stationary sensor network (induction loop detectors, microwave sensors) is costly. The goal of this work is to make accurate traffic flow estimation available in places where only PVD data is available, thus enabling the aforementioned traffic prediction solutions ubiquituously.

PVD data is obtained by sampling movement typically using GPS. This data is affected by a measurement error due to GPS accuracy and the sampling error caused by the sampling rate, i.e., not knowing where the moving object was in between position samples [39]. Map matching is needed to match tracking data to the road network (cf. [5], [28], [34], [49]. Matching the trajectories to specific road segments, the average speed [40] can easily be derived. Having large amounts of vehicles collecting such data for a given spatial area such as a city (e.g., taxis, public transport, utility vehicles, and private vehicles) allows us to create an accurate picture of the traffic condition in time and space. PVD has been extensively been used in literature to estimate travel times on road networks, e.g., [3], [11], [31], [40], [48], [52], [54], [56]. However, since PVD uses a sample of vehicles, it cannot be used directly to measure traffic flow. Further, given that the sample size varies over time and space [44], estimating traffic flow based on PVD presents a challenge.

Several studies have used different approaches to estimate traffic flow for entire road networks. Lefebvre et al. [25] use data collected from acoustic sensors; a technology introduced to overcome the prohibitive cost of stationary sensors. Current installation of such sensors are limited, compared to the wide availability of PVD data. Cellular phone data has been leveraged to estimate traffic flow [8], using the handover from one base station cell to another as an indicator of traffic flow. This approach is limited, as it may estimate the flow only at the boarders of base station cells, but not within the area of coverage of a cell. A real-time traffic flow estimation from videos taken by unmanned aerial vehicles is proposed in [21]. This approach raises privacy concerns and it is uncertain to what extent it is cost-efficient, or scalable for ubiquituous traffic flow estimation. Graph-based deep learning models have been recently proposed to capture spatial and

temporal dependencies of the road network graph, including temporal graph convolutional networks [55], spatio-temporal graph convolutional networks [51], traffic graph convolutional long short-term memory neural networks [9], dynamic spatio-temporal graph convolutional neural networks [10], attention based spatial-temporal graph convolutional networks [19], and diffusion convolutional recurrent neural networks [27]. These models still require large collections of traffic flow data for training, which is not always feasible as explained. Snowdon et al. [44] study the spatio-temporal coverage of PVD samples in relation to flow using historic data collected for the Washington DC area. This respective coverage is used to estimate the total flow based on the number of current PVD samples. The approach is limited and cannot easily be generalized as the sample size varies with time and space (roads). Relevant to our contribution, [1], [33] combine the speed estimated by PVD with the fundamental diagram to estimate traffic flow. However, directly inferring traffic flow from the average speed of sparse GPS samples leads to erroneous results, as the sample of vehicles is often a non-representative subset of the full set of vehicles on the roads. Meng et al. [31] combines data collected by static sensors and taxi trajectories in a semi-supervised learning model to infer the city-wide traffic flow. Aslam et al. [2] study the case of learning a regression model from a roving sensor network of taxi probes. However, it is not clear if taxi flow can be leveraged to generalize to general traffic flow, as taxis may be a biased sample of the population.

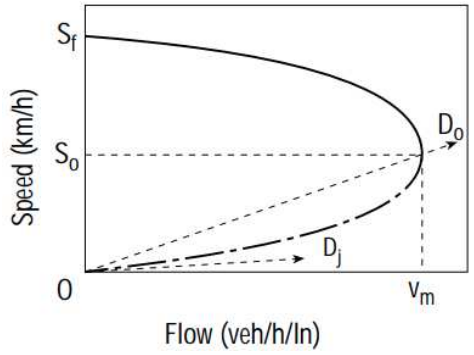


Fig. 2. Fundamental diagram depicting the speed-flow relationship, by the Highway Capacity Manual [30].

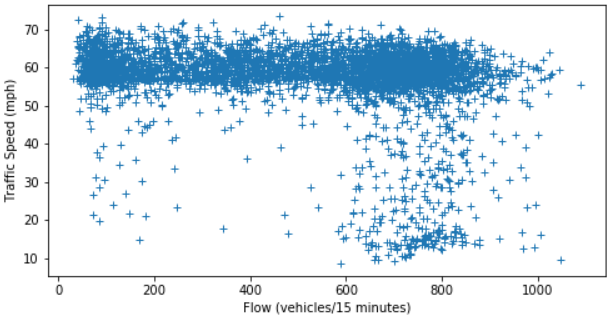


Fig. 3. Traffic speed vs. flow at the I-395 interstate (northbound) on 23 January - 20 February 2017.

III. Problem Definition

Before discussing our solution to ubiquitous traffic flow estimation, we first recapitulate the traditional definitions of the basic traffic quantities, as described by the Highway Capacity Manual [30]:

Definition 1: **Traffic Flow** is defined as the rate, per time, at which vehicles pass a point on a traffic-way.

Definition 2: **Travel Speed** is the average speed, in kilometers per hour, of a traffic stream computed as the length of a highway segment divided by the average travel time of the vehicles traversing the segment.

In the remainder of this paper, we refer to "travel speed" as "average speed", or simply as "speed".

Definition 3: **Traffic Density** is the number of vehicles on a roadway segment averaged over space, usually expressed as vehicles per kilometer.

The relationship between flow, speed and density is described by the fundamental diagram of traffic flow [30]. Figure 2 shows the relationship of speed and flow. These diagrams however, are over-simplified models of the actual relationships. For instance, Figure 3 shows the scatter-plot of real measurements of flow and speed over one week period from a sensor of I-395 northbound. We can observe that several values of flow may correspond to the same speed value, which cannot be captured by a single deterministic function like the fundamental diagram in Figure 2.

The problem that we address in this paper, is to combine globally available speed measurements with a limited number of flow time series from specific locations, in order to estimate the flow at every segment of the road network. This problem is formalized in the following.

Definition 4: (**Flow data**) Let $\mathcal{L}$ be the set of the locations of all the road segments in a road network, and let the set $\mathcal{L}_F = \{L_1, L_2, ...L_m\} \subset \mathcal{L}$ be the finite set of locations where the roadside sensors are located, also called the flow locations. Formally, the traffic flow can be expressed as a function from the domain of flow locations $\mathcal{L}_F$ and time T, to the set of positive real numbers that correspond to the rate of vehicles passing by L at that time:

$$F : \mathcal{L}_F \times T \rightarrow \mathbb{R}^+$$

Definition 5: (**Speed data**) Travel speed is a function mapping any road segment locations $\mathcal{L}$ and time T, to a positive real number that correspond to the average speed of vehicles passing from a segment at a given time:

$$S : \mathcal{L} \times T \rightarrow \mathbb{R}^+$$

We use the notations $F(L)$ and $S(L)$ to refer to the entire time series of vehicular flow and average traffic speed at a specific location L. For every flow location $L \in \mathcal{L}_F$, the time series of speed $S(L)$, and flow $F(L)$ are known. For any other location on the road network $L \in \mathcal{L} \setminus \mathcal{L}_F$, only the speed time series $S(L)$ is available. Furthermore, the coordinates of each location, the road name, direction, number of lanes and road category are known for every road segment.

Definition 6: (**Traffic Flow Estimation**) Let $L \in \mathcal{L} \setminus \mathcal{L}_F$ be the location of a segment of the road network where there is no roadside sensors. We wish to use the available time series of speed $S(L)$ at L, as well as the set of available flows $\{F(l) : l \in \mathcal{L}_F\}$ and speeds $\{S(l) : l \in \mathcal{L}_F\}$ from flow locations, to estimate the traffic flow at L.

We propose a framework that infers flow information for locations where only speed information is available. Towards this goal, the next section defines our concept of modeling prototypical road segments as so-called road segment archetypes. These archetypes are used for supervised classification of locations. Section V provides a study of different regression and estimation approaches using this framework.

IV. Road Segment Archetype-Based Traffic Flow Estimation

We combine the available speed and flow time-series from various roadside sensor locations to build models that capture the behavior of traffic flow for different parts of the network. We call these models *road segment archetypes*. This section describes our data driven approach for modeling different archetypes inspired by the classic KDD process [13]: In a preprocessing step, we first extract features from traffic speed and traffic flow time-series collected at each road-side sensor described in Section IV-A. This feature representation is used to cluster sensor locations into groups of sensor locations having similar temporal behavior in Section IV-B. For each cluster, we train a model called a *road segment archetype* that maps a sequence of observed traffic speed measurements together with time information to a traffic flow value as described in Section IV-C. Finally, in Section IV-D, we leverage these road segment archetypes to estimate traffic flow in locations where only traffic speed is known.

A. Latent Feature Extraction

We use the time-series of speed and flow measurements of each flow location to identify clusters of similar locations, with respect to their traffic behavior. These time series typically consist of a large number of measurements, each of them being a feature of the corresponding flow location. To reduce the dimensionality of our problem, we apply *Principal Component Analysis* (PCA) to extract a smaller set of *latent features* for each of the flow locations. Further technical details on this can be found online at https://github.com/olgagk/trafficFlowEstimPVD.git.

B. Clustering of Flow Locations

Using observed speed and flow measurements to assess the similarity between road segments, we group similar segments together. Each such group is represented by a generalized segment archetype. The time-series of traffic flow and speed have daily and weekly periodic patterns caused by the habitual behavior of drivers. For instance, observed patterns of commuting on roadways during weekdays typically show congestion in the early morning (in the direction towards the city center) and late afternoon (in the directions leading towards residential areas). *We aim to form groups of road segments with similar traffic patterns.* This defines a partitioning of the flow locations in $\mathcal{L}_F$ into a set of groups $\mathcal{C} = \{c_1, c_2, ..., c_n\}$, where every group $c_i \subset \mathcal{L}_F$ contains a subset of the flow locations and $c_i \cap c_j = \emptyset, \forall i \neq j$, where $c_i, c_j \in \mathcal{C}$.

To achieve the formation of flow location groups with highest similarity between the behavior of traffic characteristics between the members of each group, we *cluster* the locations of $\mathcal{L}_F$ based on the extracted latent features from both their speed and flow data. We use hierarchical agglomerative clustering of the flow locations, manually choosing a similarity threshold that yields discriminating clusters.

C. Road Segment Archetypes

Each cluster $c_i \in \mathcal{C}$ defines a group of road segments with similar traffic patterns. We combine the time-series of speed and flow from all the locations in c_i to train a flow estimation model, which we term *road segment archetype*. This archetype model M_i can be later applied on new locations with similar characteristics. These models describe the relationship between the time series of speed $S(L_j)$ with the flow $F(L_j, t)$ at any location $L_j \in c_i$ at any time t. To take into account the most recent traffic history of a road segment, our models use the speed measurements within a time window that starts at time $t - w$, ends at time t and includes a total of w measurements.

Definition 7: (**Road Segment Archetype**)
Let $S_{[t-w:t]}(L_j)$ be the set of w speed measurements at location L_j, included in the a time window $[t - w, t]$:

$$S_{[t-w:t]}(L_j) := \{S(L_j, t - w), ..., S(L_j, t)\} \tag{1}$$

Then, the road segment archetype model M_i is a regression model that estimates flow from speed for locations in $c_i \in \mathcal{C}$:

$$M_i : (\mathbb{R}^+)^w \times \mathcal{D} \times \mathcal{T} \rightarrow \mathbb{R}^+, \text{ such that:}$$

$$F(L_j, t) \sim M_i(S_{[t-w:t]}(L_j), d, \tau), \ \forall L_j \in c_i \tag{2}$$

where $\mathbb{R}^+$ is the domain of speed values, w-1 is the number of previous speed measurements that are used to take into account the most recent traffic history of a road segment, d is the day of the week (vector of 6 dummy variables), and $\tau \in \mathcal{T} = [0, \tau_{max}]$ is the time of the day. For example, $\tau_{max} = 24 \cdot 60 = 1440$ if the temporal granularity is in minutes.

Training the aforementioned model is not trivial. As discussed in the introduction, speed does not translate directly to flow (cf. Figure 3). Our hypothesis is that the dominant signal of the flow time-series is a periodic curve whose values depend mostly on the time of the day and the day of the week, as can be seen in Figure 4. On the other hand, we expect that the short-term fluctuations of flow should be more related to the behavior of traffic speed. In the following subsection, we discuss how we train a regression model using the flow and speed data of each archetype. This model is then used to estimate flow at any road segment on a given day and time,

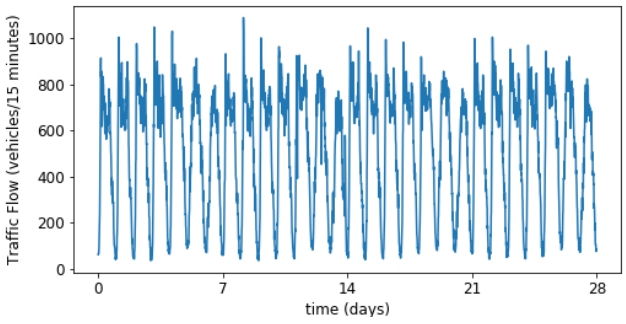


Fig. 4. Traffic Flow at a road segment of I-395 interstate.

using the average vehicular speed of this segment, as well as the day of the week and time of the day information.

We build our traffic estimation model in two phases. Phase 1 focuses on the periodic nature of traffic flow. We use the day of the week and the time of the day to get a first rough estimation of flow.

Figure 4 shows the traffic flow measurements over a period of (a) one week and (b) one month for a northbound road segment of I-395. As can be observed, each 24-hour period follows a pattern with lower flows during night time and peaks during rush hours (morning commute). Weekends differ slightly and are less busy than weekdays. These daily and weekly patterns repeat over several months in our data.

We have observed that the relationship between flow and time of the day is not linear, and in fact appears to follow a polynomial pattern. As a result, we employ polynomial regression on the time of the day.

Phase 2 focuses on the short-term variations of flow that do not necessarily follow a periodic pattern. These fluctuations can be correlated with the short-term fluctuations of traffic speed. Furthermore, short-term history of traffic patterns affects the current value of flow. Therefore we include historic values of speed within a short-term time window that ends at the current time. We also train a polynomial regression model on speed, as the relationship of speed and flow is non-linear [20].

In our implementation, during Phase 1, we first train a regression model $M_i^{(1)}$ on the flow $F(c_i)$ of all the members of a group c_i using the day of the week d and the time of the day τ of measurements as features. $F(c_i)$ is the concatenation of the time series of flow of all the road segments in c_i:

$$F(c_i) = \bigcirc_{L_j \in c_i} F(L_j)$$

Then we estimate the flow values at these known locations using the trained regression model. We calculate the difference between real and estimated flow at these locations. The result is the time series of the errors from Phase 1. We then use these errors in Phase 2 to train another regression model $M_i^{(2)}$ using $S(c_i)$, the concatenation of the time series of speed of all the road segments in c_i:

$$S(c_i) = \bigcirc_{L_j \in c_i} S(L_j)$$

Formally, the archetype model M_i of a group $c_i \in \mathcal{C}$ of Equation 2 can now be expressed as:

$$\begin{aligned} M_i(S_{[t-w:t]}(L_j), d, \tau) =& M_i^{(1)}(d, \tau) \\ &+ M_i^{(2)}(S_{[t-w:t]}(L_j)) \ , \ \forall L_j \in c_i \end{aligned} \tag{3}$$

where $M_i^{(1)}$ is the model trained during Phase 1, using only temporal features, and $M_i^{(2)}$ is the model trained in Phase 2, using traffic speed.

D. Traffic Flow Estimation

Let $L \in \mathcal{L} \setminus \mathcal{L}_F$ be the location of a segment of the road network where there are no roadside sensors. Let $\mathcal{M} = \{M_0, M_1, \cdots, M_k\}$ be the set of road archetype models. A challenge of this work is to match L with one of the road segment archetypes, using its speed information as well as its spatial characteristics (coordinates of the location, road name, direction, number of lanes, etc.). To achieve this goal, we *classify* every new location into one of the road segment archetypes (i.e., clusters) of the flow locations. In other words, every cluster c_i, which resulted from the unsupervised learning task of clustering, now becomes a class label for the supervised learning task of classification. The technical details of this step can be found in Section V-B.

Having matched l to the appropriate archetype, we can use its model $M_i \in \mathcal{M}$ to estimate the traffic flow at L.

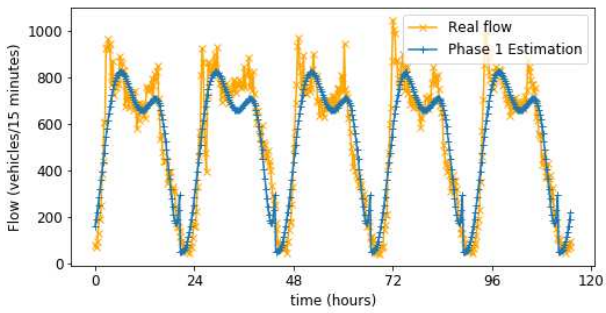
Once the new location $L \in \mathcal{L} \setminus \mathcal{L}_F$ has been classified into one of the existing groups c_i, we can apply its archetype model M_i on $S(L)$, the speed time-series of L, to estimate the traffic flow at L. This estimation consists of the sum of the results using the regression models for each of the two phases described above. Formally, if L has been classified to the road segment archetype M_i, then the estimation of its traffic flow is as follows:

$$\langle F(L,t) \rangle \sim M_i^{(1)}(d, \tau) + M_i^{(2)}(S_{[t-w:t]}(L)) \tag{4}$$

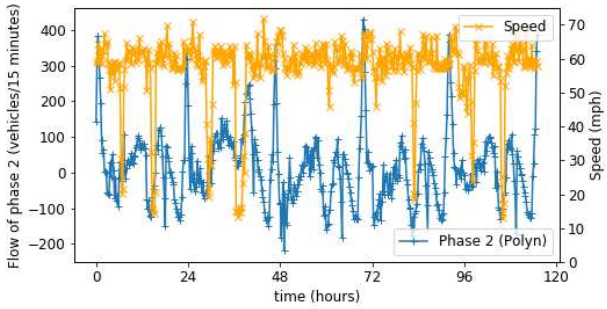
To illustrate our approach, Figure 5 presents an example of the traffic flow estimation at a road segment of I-395 northbound during January 23-27, 2017. The output of the polynomial regression of $M_i^{(1)}(d,t)$ in the first phase is shown in Figure 5(a). While it adequately approximates the periodic traffic behavior, it misses the details of the unpredictable fluctuations in speed and flow. These details are the output of $M_i^{(2)}(S_{d[t-w:t]}(L', d, t)$ based on traffic speed, as shown in Figure 5(b). The combination of these two phases provides a significantly more accurate estimation of the actual traffic flow, as can be observed in Figure 5(c).

V. Experimental evaluation

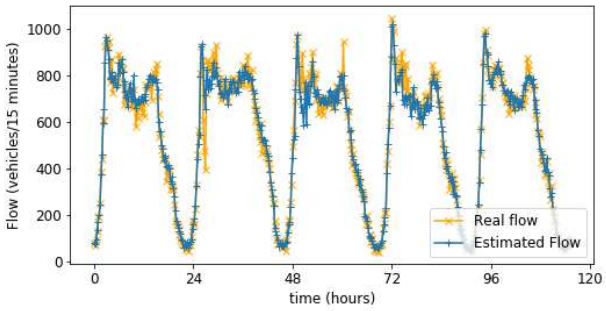
This section presents the results of our experimental evaluation on data from the Virginia Department of Transportation (VDOT). All algorithms were implemented in Python. We used implementations from the scikit-learn [38] Python package for machine learning and from the Py-earth [41] package for regression splines. The experiments were performed on a



(a) Output of Phase 1



(b) Output of Phase 2



(c) Total Estimation

Fig. 5. Estimation sample from an I-395 northbound road segment, during 23 – 27 January, 2017.

264GB-memory Intel(R) Xeon(R) 2.40GHz-CPU server running CentOS Linux. The source code of our implementation, as well as reproducibility details, can be found online at https://github.com/olgagk/trafficFlowEstimPVD.git.

A. Data

To evaluate our method, we use traffic speed and flow data. The flow data has been measured by roadside sensors deployed by the Virginia Department of Transportation (VDOT). These sensors are monitoring 36 road segments located on interstates of the northern Virginia district. Our data contains 3 months (January 01 - March 31, 2017) of measurements aggregated at 15-minute intervals, constituting a total of 17,280 features for each of the 36 sensor locations. To reduce the dimensionality of our problem, we apply *Principal Component Analysis* (PCA) to extract a smaller set of *latent features* for each

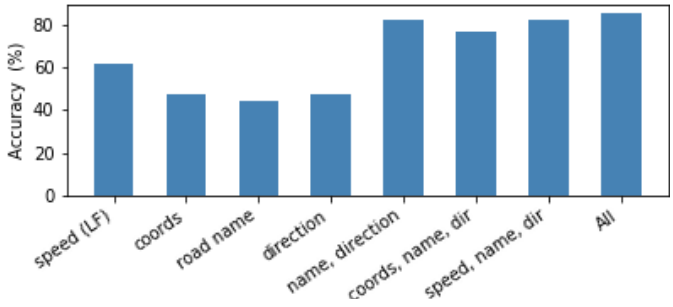


Fig. 6. Road segment classification accuracy (%) by different combinations of features.

of the sensor locations. We chose to retain the first five principal components (accounting for the most variability of the data) for the time-series of traffic flow, and the first five principal components for the speed time-series, for a total of ten features. The locations for which both traffic speed and flow measurements are available, i.e., those that belong to the set $\mathcal{L}_F$, are clustered in order to form the representative road archetype. We employ hierarchical agglomerative clustering, using Euclidean distance, to form groups of locations based on their similarity of the latent features of Speed and Flow. To obtain clusters from the resulting dendrogram, we selected a distance threshold of 0.0022 yielding 10 clusters of the 36 flow locations.

B. Classification of New Road Segments

The majority of road segments in a transportation network do not belong to $\mathcal{L}_F$, as the roadside sensors are sparsely scattered throughout the network. Thus, for those road segments, we cannot combine the information of flow and speed to assign them to groups in $\mathcal{C}$. Instead, we can use the speed, together with other known characteristics of a location. We use nearest-centroid classification and assign the label of the nearest cluster to the location that is being tested. Euclidean distance was used as the dissimilarity metric, i.e., the same metric that we had used in our clustering of flow locations. The feature vector of each location consists of the top 5 latent features of speed, as well as the coordinates of the location, the road name, direction and number of lanes. We normalize this information so that no attribute will dominate the values of the other features in the distance metric.

To evaluate the quality of the classification, we performed a leave-one-out cross validation using the 36 flow locations. The original cluster id of a location is used as the ground truth for the location's class label. The results of our analysis are shown in Figure 6. Using only the speed information for classification yields unsatisfactory results; only 61.7% of the sensors were classified correctly to their original group. Using only the location information (spatial coordinates) results in an even lower classification accuracy of 47.1%. Classifications based only on road name or segment direction yield accuracy of 44% and 47.1% respectively. Combining multiple features increases the overall accuracy. The highest classification accuracy was achieved by combining the information of speed (i.e., the 5

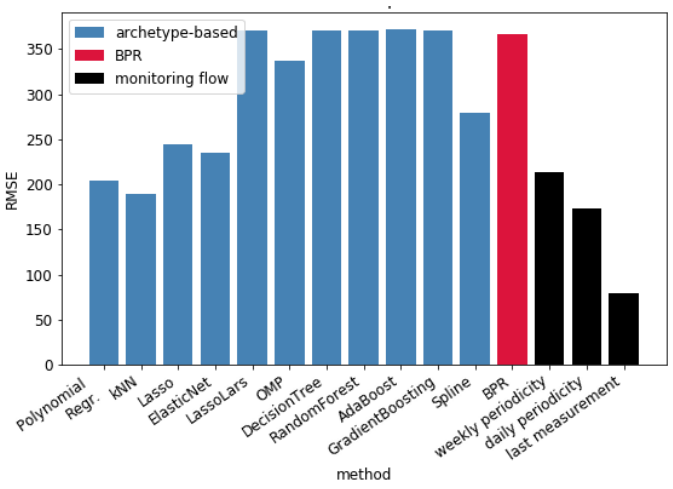


Fig. 7. Root Mean Square Error (RMSE) of the tested models. The proposed approach (archetype-based) is shown in blue. The BPR function based estimation is shown in crimson red. Three periodicity-based approaches are shown in black assuming available historic traffic flow data.

latent features from applying PCA on the speed time series $S(L)$), spatial coordinates of the road segment, direction and road name. The resulting accuracy is 85.29%, which is a strong result considering that this is a 10-class classification problem, where random guessing would yield an expected accuracy of 10%.

C. Traffic Flow Estimation Approaches

The approaches that we tested in our experiments include the proposed road segment archetype based estimation, the flow estimation based on the BPR formula [7], and a set of periodicity based approaches, which we describe below.

Archetype-based approach. To evaluate our road segment archetype based approach, we performed a leave-one-out cross validation. For each experiment, we consider one of the flow locations $l \in \mathcal{L}_F$ to be an 'unknown' road segment where only the travel speed is known. The remaining 35 flow locations in $\mathcal{L}_F \setminus \{l\}$ are the 'known' flow locations where both speed and flow information is available. We cluster these 35 road segments into groups and classify the remaining 'unknown' road segment l into one of these groups. Let g be the group to which l was classified. We then train a wide range of regression models (listed below), using the speed and flow information of the selected group g. We use these road segment archetype models to predict the traffic flow at l, using the available speed $S(l)$, day of the week and time of the day information. We compare the estimated values of flow $\langle F(l) \rangle$ with the ground truth $F(l)$, i.e., the actual flow at l. We report our results of the average $RMSE$ and R^2-score, averaged over all the tested road segments, in Figures 7 and 9 respectively.

We experimented with a variety of models to test their efficiency in traffic flow estimation, using our road segment archetype-based approach. This includes a set of Generalized Linear Models (GLMs), as well as non-linear models, such as k-nearest neighbor regression, and decision trees. The models that we tested are the following:

- **Polynomial regression** fits a polynomial function minimizing the sum of square residuals.
- **Lasso regression** [47] constrains the sum regression coefficients, reducing the number of variables of the problem.
- **Elastic Net regression** [57] estimates sparse coefficients like Lasso, while maintaining the regularization properties of Ridge.
- **LARS Lasso** [12] is a Lasso model implemented using the least angle regression algorithm.
- **Orthogonal Matching Pursuit (OMP)** [37] is a recursive forward feature selection algorithm that approximates the fit of a linear model, based on the matching pursuit method [29].
- **Multivariate Adaptive Regression Splines** [15] is a non parametric approach using regression splines.
- **Nearest Neighbors Regression (kNN)** predicts the inverse-distance weighted average the k nearest neighbors of a point.
- **Decision Tree regression** is a non-parametric model that learns decision rules for prediction.
- **AdaBoost** [14] is a boosting algorithm for ensemble models using the weighted average multiple other models.
- **Random forests** [6] are ensemble models consisting of decision trees, each built from a subset of the training set.
- **Gradient Boosting** [16] is a boosting generalization that allows for optimization of arbitrary differentiable loss functions.

After experimenting using w=2 to 15 values of historic speed in Phase 2, and with polynomial features of up to 9th degree, we selected (i) w=10, and (ii) polynomial features of up to 7th degree in both phases, which gave the highest cross validation accuracy.

BPR. We compare our results against the BPR formula [7] that describes the relationship between traffic speed and flow, given by the following equation,

$$S = \frac{s_0}{(1 + \alpha \cdot \left(\frac{F}{c}\right)^{\beta})} \tag{5}$$

where S is the speed, F is the flow, c is the capacity of the road segment, s_0 is the free-flow speed, and α, β are road parameters. Typical values for α vary from 0 to 1.0 and for β from 4 to 11 [20], [43]. Solving for traffic flow, Equation 5 becomes:

$$F = c \cdot \left(\frac{1}{\alpha} \cdot (\frac{s_0}{S} - 1)\right)^{-\beta}$$

For each road segment, we first optimize the parameters α and β, given the actual speed and flow measurements, and the capacity of the segment. As free-flow speed, we used the maximum speed limit. We use the real speed values to estimate the flow. Note that in reality, when traffic flow is unknown, no such initial optimization of the parameters can

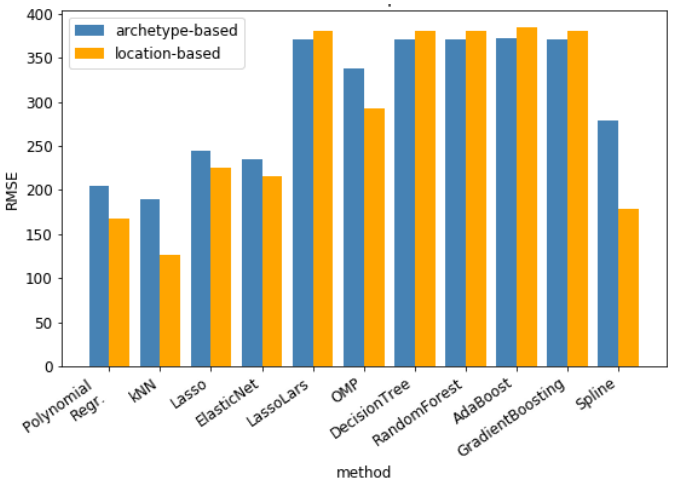


Fig. 8. Root Mean Square Error (RMSE) of the proposed approach (archetype-based), shown in blue, vs. an alternative location-based approach (orange) which requires traffic flow data measured at every location.

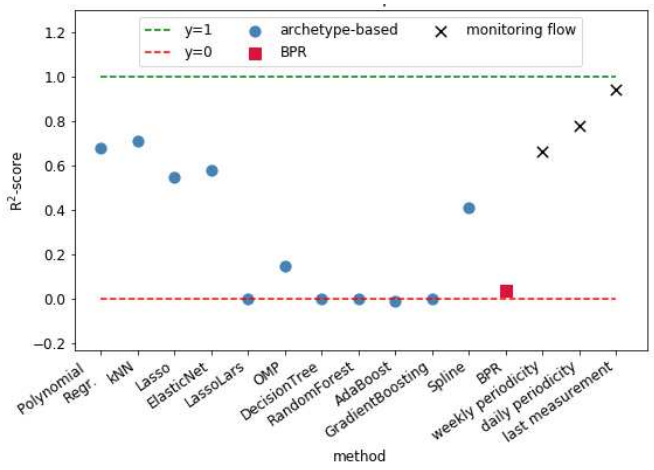


Fig. 9. R^2 score of the tested models. BPR (red square) and the (unrealistic) periodicity-based approaches (black crosses) are included in both graphs.

be performed. Instead, the values for α, β and road capacity of each road segment would be approximated or looked up in a reference manual. Thus, our results present the best-case estimation accuracy of the BPR formula.

Periodicity-based approaches. For the same reasons as above, we also include three periodicity-based estimations: (i) the *weekly* periodicity that estimates a flow value as the traffic flow of the same road segment one week ago (at the same day and time), (ii) the *daily* periodicity that estimates a flow value as the traffic flow of the same road segment 24 hours ago, and (iii) the *last measurement* that estimates flow as the last measurements that was taken 15 minutes ago from the same sensor.

All these three approaches require past traffic flow values, which is impractical given the general lack of flow data for a network. However, our proposed archetype-based approach compensates for this lack of data and provides excellent flow estimates as evidenced by this "unfair" comparison.

1) Estimation Quality: **Road Archetypes vs BPR error.** The accuracy of our estimation is shown in Figures 7-9 in blue color. The root mean square error of all the tested methods is depicted in Figure 7. The highest accuracy is achieved by the k-Nearest Neighbor regression, followed by the Polynomial regression (i.e., the Linear regression model using polynomial features) with a difference of about 7%. kNN manages to achieve about half (51%) of the RMSE of the corresponding BPR estimation, even though the latter is based on the real flow of the road segment rather than the archetype flow. It even beats the weekly periodicity and is comparable to the daily periodicity approach, even though these approaches use known past flow measurements at the location of interest, which our approach does not use neither for the training phase, nor for the estimation. The Lasso, Elastic Net, and Spline regressions also achieved adequate results, with errors that are less than 10% of the road capacity. As expected, the last measurement of flow is actually a very good estimate of the traffic flow at the same segment 15 minutes later. However, this approach would require a continuous monitoring of flow everywhere in the road network, which is in reality not available.

Archetype-based vs Location-based errors. To assess to what degree our road archetype based models provide meaningful results, we perform an additional set of experiments, in which we train the regression models using the original traffic flow and speed of a road segment $l \in \mathcal{L}_F$ as training data, instead of performing the archetype modelling of Section IV-C. We then use this model to estimate the traffic flow at the same segment l. To perform this experiment, we use the first 1.5 month of the time-series as training and the next 1.5 month as the test set. Figure 8 shows a comparison of our archetype based approach to the location based approach, for each of the models. The location based approaches are shown in orange color (to the right of each corresponding archetype-based estimation error) for comparison. It must be stressed that these results would only be possible if there existed a recording of flow measurements over time. As explained in Section I this is infeasible in practice due to limited sensor coverage. We only include this result to show that our archetype-based flow estimation at an 'unknown' road segment can give comparable results to those estimations that are based on known previous traffic flow measurements of the road segment. It must be noted that while using the real flow in the training phase of these models yields better results, the difference is still manageable, 33.09% for kNN, 18.07% for polynomial regression, and only 7.4% for Lasso. The worst case is the 36% difference for Splines. On the other hand, the decision tree based models and ensemble methods performed on average 3% better using the road archetypes rather than the original historic flow of the road segment.

Coefficient of determination. The coefficient of determination (R^2-score) is reported in Figure 9. The blue dots correspond to the models trained by the road archetype speed

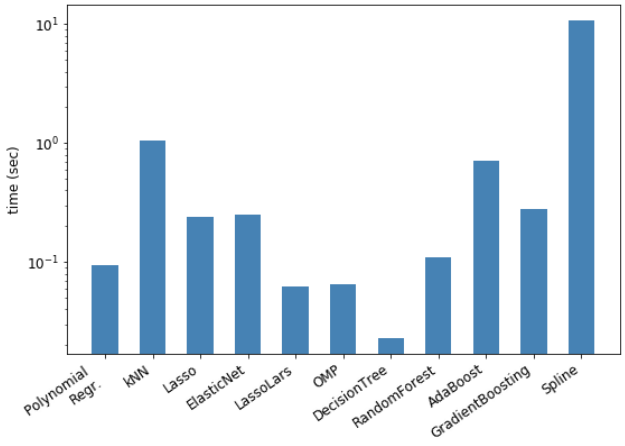


Fig. 10. Running time of the tested models. The preprocessing time is not included for the model comparison.

and flows, while the red square corresponds to BPR, and x shows the periodicity-based approaches. An R^2-score of 1 (green dotted line) implies a perfect regression model, wheras an R^2-score of zero (red dotted line) corresponds to the quality of a model that predicts a constant mean flow of a segment. Negative values correspond to poor quality, and there is no lower limit as a model can be arbitrarily bad. Given that we assume that we do not know the true traffic flow, any value of 0 or higher is in reality a very positive result by our approach. kNN, Polynomial, Elastic Net, Lasso, Spline and OMP regression all score much higher than the BPR estimation, while Lasso Lars, decision tree, random forest, Ada boost and gradient boosting models still achieve somewhat acceptable performance. kNN regression is the winner with an R^2-score of 0.71, which is 17.8 times better than the BPR, and only 18.2% lower than the corresponding kNN trained on the true traffic flow. We remind the reader that the weekly and daily periodicity, as well as the"last measurement" approaches are only applicable for the scenario where both traffic speed and flow measurements of each road segment are available (location based approach), whereas they can not be used if no flow measurements are being monitored at a road segment. The latter is the case for the majority of road segments, due to the high cost of installation and maintenance of the monitoring equipment. Thus, these methods, despite their high accuracy, cannot be used to estimate traffic for the largest part of the road network.

Overall, the experiments conclusively show that the proposed road segment archetype method is a reliable traffic flow estimation method for an entire road network with sparse sensor coverage.

2) Running time: We report the total running time for training and testing each of the regression models, using our archetype based approach in Figure 10. Splines can be computationally expensive compared to simpler models like lazy k-Nearest Neighbor regression, or Polynomial regression, that also proved to be more efficient in terms of estimation quality. In our experiments, kNN regression required just over a second to run, which is tolerable since it also gives the lowest estimation errors. Lasso and Elastic Net run within 240 miliseconds. Polynomial regression finished in 94 miliseconds and is a very good candidate for cases when running time is as important as the accuracy of the result. The fastest model was the decision tree with a running time just below 23 miliseconds with a lower, but acceptable, flow estimation quality.

VI. Conclusions

Given the advent of novel sensor technology to estimate traffic conditions, this so-called probe vehicle data provides us with speed information for the entire road network, but no traffic flows. This work leverages sparse stationary sensors and novel PVD data to estimate flow for the entire road network. Parts of the network with similar traffic patterns were combined to model Road Segment Archetypes, which are then used to estimate the traffic flow for the entire network. Experimental evaluation of a wide variety of regression models and using real-world traffic data shows that road segment archetypes provide better estimates than existing methods. Moreover, the estimates are comparable to hypothetical flow-based estimates. This is a strong result, given that in our approach the models never observe any instance of the real flow at the location of interest. Directions for future work are to create a flow map for the entire road network and to empirically verify the results with the respective stakeholders.

Acknowledgements

This research has been supported by the National Science Foundation grant "AitF: Collaborative Research: Modeling movement on transportation networks using uncertain data" NSF-CCF 1637541. We would like to thank the Virginia Department of Transportation for providing us with their data.

References

[1] K. A. Anuar, F. G. Habtemichael, and M. Cetin. Estimating traffic flow rate on freeways from probe vehicle data and fundamental diagram. In *IEEE ITSC 2015*, pages 2921–2926, 2015.

[2] J. A. Aslam, S. Lim, X. Pan, and D. Rus. City-scale traffic estimation from a roving sensor network. In *ACM SenSys*, pages 141–154, 2012.

[3] A. Bhaskar, E. Chung, and A. Dumont. Fusing loop detector and probe vehicle data to estimate travel time statistics on signalized urban networks. *Comp.-Aided Civil and Infrastruct. Engineering*, 26(6):433–450, 2011.

[4] J. Black. *Urban transport planning: Theory and practice*. Routledge, 2018.

[5] S. Brakatsoulas, D. Pfoser, R. Salas, and C. Wenk. On map-matching vehicle tracking data. In *VLDB'05*, pages 853–864. VLDB Endowment, 2005.

[6] L. Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[7] Bureau of Public Roads. Traffic assignment manual. *Department of Commerce, Urban Planning Division, Washington, DC, USA*, 1964.

[8] N. Caceres, L. M. Romero, F. G. Benitez, and J. M. del Castillo. Traffic flow estimation models using cellular phone data. *IEEE TITS*, 13(3):1430–1441, 2012.

[9] Z. Cui, K. Henrickson, R. Ke, and Y. Wang. Traffic graph convolutional recurrent neural network: A deep learning framework for network-scale traffic learning and forecasting. *IEEE Transactions on Intelligent Transportation Systems*, 2019.

[10] Z. Diao, X. Wang, D. Zhang, Y. Liu, K. Xie, and S. He. Dynamic spatial-temporal graph convolutional neural networks for traffic forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 890–897, 2019.

[11] A. Efentakis, S. Brakatsoulas, N. Grivas, G. Lamprianidis, K. Patroumpas, and D. Pfoser. Towards a flexible and scalable fleet management service. In *IWCTS@SIGSPATIAL*, pages 79–84, 2013.

[12] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani. Least angle regression. *Ann. Statist.*, 32(2):407–499, 04 2004.

[13] U. Fayyad, G. Piatetsky-Shapiro, and P. Smyth. The kdd process for extracting useful knowledge from volumes of data. *Communications of the ACM*, 39(11):27–34, 1996.

[14] Y. Freund and R. E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of computer and system sciences*, 55(1):119–139, 1997.

[15] J. H. Friedman. Multivariate adaptive regression splines. *The annals of statistics*, pages 1–67, 1991.

[16] J. H. Friedman. Greedy function approximation: a gradient boosting machine. *Annals of statistics*, pages 1189–1232, 2001.

[17] M. C. Gonzalez, C. A. Hidalgo, and A.-L. Barabasi. Understanding individual human mobility patterns. *nature*, 453(7196):779, 2008.

[18] B. Greenshields, W. Channing, H. Miller, et al. A study of traffic capacity. In *Highway research board proceedings*, volume 1935. National Research Council (USA), Highway Research Board, 1935.

[19] S. Guo, Y. Lin, N. Feng, C. Song, and H. Wan. Attention based spatial-temporal graph convolutional networks for traffic flow forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 922–929, 2019.

[20] Highway Capacity Manual. Highway research board. *Special Report*, 87:1965, 1965.

[21] R. Ke, Z. Li, S. Kim, J. Ash, Z. Cui, and Y. Wang. Real-time bidirectional traffic flow parameter estimation from aerial videos. *IEEE TITS*, 18(4):890–901, 2017.

[22] B. S. Kerner. Three-phase traffic theory and highway capacity. *Physica A: Statistical Mechanics and its Applications*, 333:379–440, 2004.

[23] J. Kwon, P. Varaiya, and A. Skabardonis. Estimation of truck traffic volume from single loop detectors with lane-to-lane speed correlation. *Transportation Research Record: Journal of the Transportation Research Board*, pages 106–117, 2003.

[24] S. Lee and D. Fambro. Application of subset autoregressive integrated moving average model for short-term freeway traffic volume forecasting. *Transportation Research Record: Journal of the Transportation Research Board*, pages 179–188, 1999.

[25] N. Lefebvre, X. Chen, P. Beauseroy, and M. Zhu. Traffic flow estimation using acoustic signal. *Eng. Appl. of AI*, 64:164–171, 2017.

[26] Y. Li and C. Shahabi. A brief overview of machine learning methods for short-term traffic forecasting and future directions. *SIGSPATIAL Special*, 10(1):3–9, 2018.

[27] Y. Li, R. Yu, C. Shahabi, and Y. Liu. Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. In *International Conference on Learning Representations*, 2018.

[28] Y. Lou, C. Zhang, Y. Zheng, X. Xie, W. Wang, and Y. Huang. Map-matching for low-sampling-rate GPS trajectories. In *ACM SIGSPATIAL 2009*, pages 352–361, 2009.

[29] S. Mallat and Z. Zhang. Matching pursuits with time-frequency dictionaries. *IEEE Transactions on Signal Processing*, 41(12):3397–3415, 1993.

[30] H. C. Manual. Highway capacity manual. *Washington, DC*, 2, 2000.

[31] C. Meng, X. Yi, L. Su, J. Gao, and Y. Zheng. City-wide traffic volume inference with loop detector data and taxi trajectories. In *ACM SIGSPATIAL 2017*, pages 1:1–1:10, 2017.

[32] L. Muñoz, X. Sun, R. Horowitz, and L. Alvarez. Traffic density estimation with the cell transmission model. In *American Control Conference 2003*, volume 5, pages 3750–3755. IEEE, 2003.

[33] T. Neumann, P. L. Bohnke, and L. C. T. Tcheumadjeu. Dynamic representation of the fundamental diagram via bayesian networks for estimating traffic flows from probe vehicle data. In *IEEE ITSC 2013*, pages 1870–1875, 2013.

[34] P. Newson and J. Krumm. Hidden markov map matching through noise and sparseness. In *ACM SIGSPATIAL 2009*, pages 336–343. ACM, 2009.

[35] I. Okutani and Y. J. Stephanedes. Dynamic prediction of traffic volume through kalman filtering theory. *Transportation Research Part B*, 18(1):1–11, 1984.

[36] B. Pan, U. Demiryurek, and C. Shahabi. Utilizing real-world transportation data for accurate traffic prediction. In *ICDM 2012*, pages 595–604. IEEE, 2012.

[37] Y. C. Pati, R. Rezaiifar, and P. S. Krishnaprasad. Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition. In *Signals, Systems and Computers, 1993. 1993 Conference Record of The Twenty-Seventh Asilomar Conference on*, pages 40–44. IEEE, 1993.

[38] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[39] D. Pfoser and C. S. Jensen. Capturing the uncertainty of moving-object representations. In *SSD '99: Proceedings of the 6th International Symposium on Advances in Spatial Databases*, pages 111–132. Springer-Verlag, 1999.

[40] D. Pfoser, N. Tryfona, and A. Voisard. Dynamic Travel Time Maps - Enabling Efficient Navigation. In *SSDBM*, pages 369–378. IEEE Computer Society, 2006.

[41] J. Rudy. Package name sklearn-contrib-py-earth version 0.1. 0.

[42] X. Shan, P. Hao, X. Chen, K. Boriboonsomsin, G. Wu, and M. J. Barth. Vehicle energy/emissions estimation based on vehicle trajectory reconstruction using sparse mobile sensor data. *IEEE TITS*, 20(2):716–726, 2018.

[43] R. Singh. Beyond the bpr curve: Updating speed-flow and speed-capacity relationships in traffic assignment. In *5th National Conference on Transportation Planning Methods Applications-Volume II*, 1995.

[44] J. Snowdon, O. Gkountouna, A. Züfle, and D. Pfoser. Spatiotemporal traffic volume estimation model based on gps samples. In *GeoRich@ACM SIGMOD, 2018*, pages 1–6. ACM, 2018.

[45] C. Song, Z. Qu, N. Blumm, and A.-L. Barabási. Limits of predictability in human mobility. *Science*, 327(5968):1018–1021, 2010.

[46] G. S. Thakur, P. Hui, and A. Helmy. Modeling and characterization of urban vehicular mobility using web cameras. In *IEEE INFOCOM 2012*, pages 262–267, 2012.

[47] R. Tibshiranit. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B*, 58(1):267–288, 1996.

[48] Y. Wang, Y. Zheng, and Y. Xue. Travel time estimation of a path using sparse trajectories. In *ACM SIGKDD 2014*, pages 25–34, 2014.

[49] C. Wenk, R. Salas, and D. Pfoser. Addressing the need for map-matching speed: Localizing globalb curve-matching algorithms. In *SSDBM*, pages 379–388. IEEE Computer Society, 2006.

[50] D. Wilkie, J. Sewall, and M. C. Lin. Flow reconstruction for data-driven traffic animation. *ACM Trans. Graph.*, 32(4):89:1–89:10, 2013.

[51] B. Yu, H. Yin, and Z. Zhu. Spatio-temporal graph convolutional networks: a deep learning framework for traffic forecasting. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, pages 3634–3640, 2018.

[52] J. Yuan, Y. Zheng, C. Zhang, W. Xie, X. Xie, G. Sun, and Y. Huang. T-drive: Driving directions based on taxi trajectories. In *ACM SIGSPATIAL*, GIS '10, pages 99–108, 2010.

[53] X. Zhan, R. Li, and S. V. Ukkusuri. Lane-based real-time queue length estimation using license plate recognition data. *Transp. Research Part C*, 57:85–102, 2015.

[54] X. Zhan, S. V. Ukkusuri, and C. Yang. A bayesian mixture model for short-term average link travel time estimation using large-scale limited information trip-based data. *Automation in Construction*, 72:237–246, 2016.

[55] L. Zhao, Y. Song, C. Zhang, Y. Liu, P. Wang, T. Lin, M. Deng, and H. Li. T-gcn: A temporal graph convolutional network for traffic prediction. *IEEE Transactions on Intelligent Transportation Systems*, 2019.

[56] F. Zheng and H. Van Zuylen. Urban link travel time estimation based on sparse probe vehicle data. *Transportation Research Part C*, 31:145–157, 2013.

[57] H. Zou and T. Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society Series B*, 67(2):301–320, 2005.