Characterizing Taxi Flows in New York City

Marjan Momtazpour
Discovery Analytics Center
Department of Computer Science
Virginia Tech, Blacksburg, VA
marjan@cs.vt.edu

Naren Ramakrishnan
Discovery Analytics Center
Department of Computer Science
Virginia Tech, Arlington, VA
naren@cs.vt.edu

ABSTRACT

We present an analysis of taxi flows in Manhattan (NYC) using a variety of data mining approaches. The methods presented here can aid in development of representative and accurate models of large-scale traffic flows with applications to many areas, including outlier detection and characterization.

Categories and Subject Descriptors

H.2.8 [**Database Applications**]: [Data mining - Spatial databases and GIS]; G.2.2 [**Graph Theory**]: Network problems

General Terms

Algorithms, Experimentation

Keywords

Data mining, urban computing, dynamic network analysis, clustering, role dynamics, outlier detection.

1. INTRODUCTION

The rapid growth in urban populations has highlighted the importance of harnessing data-driven methods to aid in city planning, including in areas like stemming air pollution, controlling energy consumption, and relieving traffic congestion [32]. Modern datasets from wireless sensor networks can aid in understanding traffic flows at a scale hitherto unrealized.

One of the main concerns in large urban cities is to analyze traffic flows with a view toward characterizing both regularities and anomalies; detection of anomalies (e.g., caused by accidents, protests, sports, celebrations, disasters) for instance can be utilized to help mitigate congestion and diagnose bottlenecks. In this paper, we analyze taxi trips in New York City logged in 2013. Our goals are to infer knowledge about the pattern of locations w.r.t their profiles, to find hotspot locations, and to detect and track anomalies over time. Using clustering approaches, we categorize the city into smaller units and characterize locations based on their daily taxi pick-up and drop-off demands. We also develop a novel probabilistic graph representation of traffic flow to infer common profiles of each location. We investigate locations by extracting their local and egonet features in the traffic flow graph. Then, we extract the role of each location in graph using role extraction methods and finally detect spatio-temporal outliers using the extracted roles.

Our contributions are thus:

- Developing a novel average probabilistic flow graph to capture the behavior of traffic flow in each location.
- Characterizing interesting locations using anomaly detection methods applied over the average probabilistic flow graph.
- Using role extraction and role-change detection to understand normal roles of each location and find spatio-temporal anomalies in dynamic graphs.

UrbComp '15, August 10, 2015, Sydney, Australia.

2. RELATED WORK

Taxi Datasets: Taxis do not have pre-specified routes and schedules and can provide a unique insight into the mobility of people through a city. There have been several works on mining taxi GPS traces in three main categories: social dynamics, traffic dynamics, and operational dynamics [5]. In social dynamics, the behavior of a group of people is studied for several purposes such as to identify hotspots, to characterize locations based on functionality, and to find frequent trajectories and connectivity (linkage) between regions. The results are useful as a guide to future decision making [5]. In traffic dynamics, the dynamics of congestion levels of vehicles have been studied. For example, travel time and speed or adverse traffic events or even air pollution resulting from traffic can be analyzed [5]. As an example, Wang et al. [27] estimate travel time of a path in a sparse trajectory dataset using tensors. In operational dynamics, the goal is to provide useful information to drivers (and passengers). Ranking drivers, taxi-finding strategies [31], taxi ride-sharing, route planning, anomaly detection (accident, road repair, fraudulent drivers), and route prediction (travel time estimation) are a few tasks that have been studied under this category [5].

Graph Mining: One of the more popular graph mining techniques applied here (e.g., to bike usage data) is community clustering. In [8], the authors clustered bike-sharing

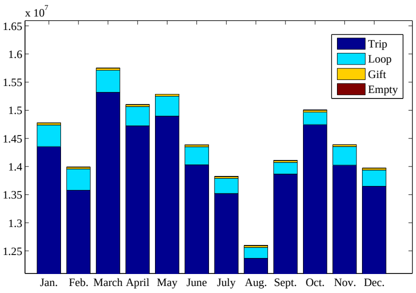


Figure 1: Noisy data vs. valid data.

stations according to their usage profiles using Poisson mixtures. Bike demand for NYC bikes has been analyzed in [25]. The authors in [3] used community detection with a gravity model applied to bicycle flow dataset. Cheng et al. [7] used an ARIMA model to capture autocorrelation in road traffic data locally and dynamically. Min et al. [17] used a hybrid spatio-temporal method to forecast traffic also using an ARIMA model. The authors of [28] used spatio-temporal random effect models to predict traffic flows.

Liu et al. [16] investigate inter-urban movements from a check-in dataset to analyze the underlying patterns of trips and spatial interactions by fitting gravity models. For visualization, the authors in [34] proposed a flow clustering method to cluster flows in order to avoid cluttering while revealing abstracted flow patterns. In order to estimate missing data or to find structure of different units in different problem domains, matrix factorization and tensor decomposition have been studied previously [27]. As an example, Lian et al. [14] exploit weighted matrix factorization to view mobility records in location-based social networks for point-of-interest recommendation. As another example, in [33], noise situations in NYC are modeled using tensor decompositions. In [9], a tensor is created for traffic flow state and clustering methods are used to derive traffic states.

Anomaly Detection: Liu et al. [15] proposed an algorithm to construct outlier causality trees based on spatio-temporal properties of detected outliers. Chawla et al. [6] proposed a two-step mining framework to infer the root causes of anomalies in road traffic data. Shafer et al. [24] proposed a novel approach using Kalman filtering as a state estimation model for mining large bursty time series and to find trends and anomalies. Xu et al. [29] identify and rank crossroads in a road network using a tripartite graph. Finally, the authors in [11] find all road segment outliers which have different traffic load than their expected values.

There is a rich literature of methods for anomaly detection in graphs [12]. For instance, Akoglu et al. [1] find anomalies using egonet features of the graph. According to [1], anomalies can be of different types including near-cliques, stars, heavy vicinities, and dominant heavy links. In another work, a non-negative residual matrix factorization method has been proposed to detect anomalies in graphs [26]. An ensemble of different methods to detect anomalies in dynamic graphs is proposed in [20]. A detailed survey on anomaly detection in graphs can be found in [2]. Furthermore, various methods used for anomaly detection in dynamic graphs have been reviewed in [19].

3. PRELIMINARY ANALYSIS

3.1 Dataset Description and Pre-processing

The dataset contains Yellow cab trips of NYC in 2013 (raw size ~ 45GB) which is publicly available*. Information such as pick-up and drop-off geographical coordinates as well as time, distance, and price of trips have been logged in this dataset. The total number of trips is 173,179,759. A pre-processing step has been performed to remove missing values and noisy data such as invalid geographical coordinates, loops (trips with the same pick-up and drop-offs), gifts (trips with zero traveled distance but with registered payment), and trips with no passenger(s). The portion of invalid data compared to the valid part is shown in Fig. 1.

Since Manhattan is one the most complex and highly populated urban areas in the world, we focused on Manhattan. For simplicity, we considered a rectangular area, where latitudes are bounded between 40.7 and 40.85, and longitudes are between -74.02 and -73.90. Total trips made within Manhattan after noise removal is 143,329,066 in 365 days.

3.2 City Decomposition

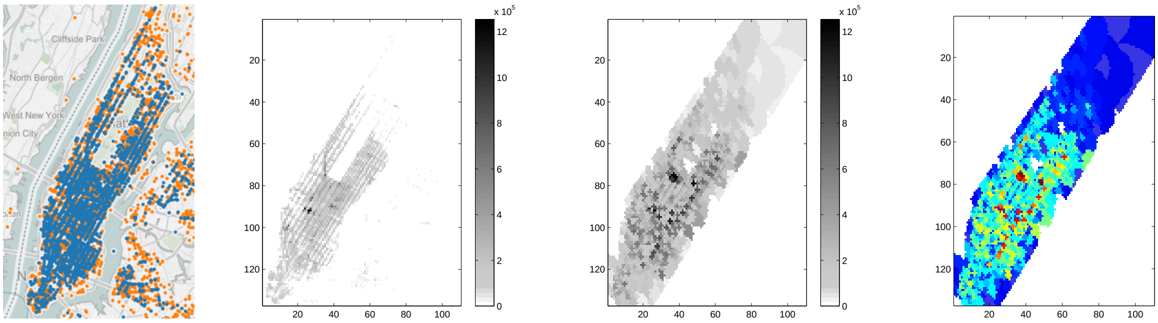
One simple and popular approach to split up the city into blocks is grid decomposition. We divide Manhattan into predetermined equal-sized blocks to logical blocks of Manhattan. However, due to the high number of vacant blocks and also behavioral similarity of adjacent ones, clustering adjacent blocks is recommended [5]. Various techniques have been used for this purpose. As an example, Cao et al. [4] used a spatial clustering algorithm to analyze composition of cities in terms of their functional behavior. They used k-means via PCA to cluster individuals into groups. Here, we applied hierarchical spatial clustering on less populated blocks (blocks with less than 200 pick-ups/drop-offs per day). The number of blocks after clustering reduced from 15,070 to 1,204 clusters.

The population of trips in terms of the number of pick-ups and drop-offs is illustrated in Fig. 2. Fig. 2(a) shows an example of the exact latitude and longitude of pickups and drop-offs in Manhattan at 5am on March 3rd, 2013. The total number of pickups and drop-offs for each block before clustering is shown in Fig. 2(b). Population of each group after clustering is shown in Fig. 2(c). For a better visualization, the differentiation between clusters is illustrated in Fig. 2(d). It should be noted that the total number of pick-ups and drop-offs are calculated over the year.

3.3 Location Characterization

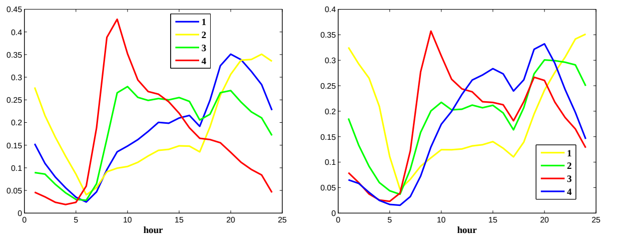
Discovering functional regions (residential, business, etc.) or categorizing people (student, workers, etc.) is valuable for city planners to comprehend activities of individuals and decide on the placement of new infrastructures [32, 13, 18]. We calculate the average daily demand to find communities with similar daily activity. K-spectral centroid (KSC) clustering is a time series clustering method which deploys a similarity metric that is scale and shift invariant. We use KSC-clustering with initial clusters driven using k-means (k=4) on averaged pick-ups and drop-offs. An adaptive wavelet-based incremental approach of this algorithm is used for this purpose [30]. Clustering results for pick-up profiles are shown in Fig. 3(a) and the geographical areas related to each curve are shown in Fig. 4(a). The corresponding results for drop-offs are shown in Fig. 3(b) and Fig. 4(b). As

*http://www.andresmh.com/nyctaxitrips/



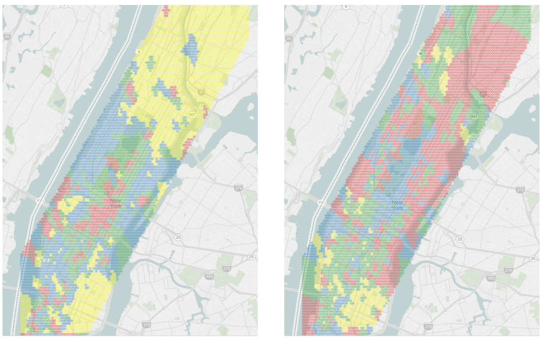
(a) (b) (c) (d)

Figure 2: (a) Map of Manhattan with pick-ups (blue) and drop-offs (orange) at 5am on March 3rd, 2013. (b) Total number of pick-ups and drop-offs for each block in grid. (c) Population of each cluster. (d) Clusters of blocks.



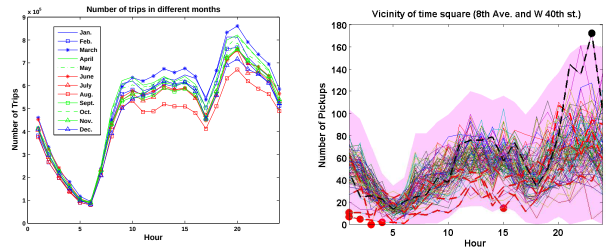
(a) (b)

Figure 3: Clustering locations based on their daily profile at (a) pick-up, and (b) drop-off.



(a) (b)

Figure 4: Clustered areas based on their daily profile on (a) pick-up, and (b) drop-off.



(a) (b)

Figure 5: (a) Number of trips at each month at each hour. Blue, green, and red colors represents cold, medium, and hot temperatures. (b) Example of traffic bursts (black) and absenteeism (red) in the number of pick-ups in one location. Each curve represent one day of the year. The shaded pink area shows the normal range of variation.

these figures depict, yellow-colored areas are the ones with higher activities during the night (7pm-6am). The north side of Manhattan has higher activities in terms of pick-up demands while the southeastern part of Manhattan has higher activities in both pick-ups and drop-offs. Red colored areas have high demand in the morning (8am-10am) and red colored drop-off locations are indicators of the business areas of the city. Other than characterization of functionality of locations, these results are helpful for recommendation on where to pickup a passenger or to find the taxis.

The total number of trips per each month at each hour is shown in Fig. 5(a). Months are clustered according to their temperature values in 2013. Red, green, and blue indicate warm, mild, and cold temperatures, respectively. As this figure depicts, the demand during colder months is higher. According to the above observations, for simplicity, in the rest of the paper we divide the 24 hours in a day into four 6-hour time slots.

4. TRAFFIC FLOW GRAPH

In order to derive an abstract explanation of behavior of people, we create an average network graph of transportation flows. This average graph helps us to understand the normal behavior of taxi transportation in the city. Following the construction of such a graph, at each timestamp, we compare the traffic graph for that instant with the averaged graph to detect anomalies. Anomalies will thus denote areas that have characteristics distinct from the average graph. For these purposes, after extracting local and egonet features of the graph, we employed two graph mining methods to explain the dynamics of traffic flow. The first method is to extract network signatures of locations using power law relations (as proposed in [1]). The other method is to use role extraction methods to understand the structural behavior of nodes and to detect outliers by finding significant changes in role memberships.

4.1 Graph Model and Feature Extraction

In our graph model, we assume each node is a clustered area (from Section 3.2) and each edge represents the existence of at least one trip between two areas. Hence, the graph will be a directed one and the weight of an edge shows the number of trips. It should be noted that at different times, the dynamics of the graph changes in terms of the existence and weight of an edge, not in terms of the number of nodes.

Definition 1. Traffic flow graph of taxis at time t of day d is denoted by $\boldsymbol{G}_t^d(\boldsymbol{V}, \boldsymbol{E}_t^d)$, where $\boldsymbol{V} = \{v_1, v_2, ..v_n\}$ is a set of nodes. Each node, v_i, in this graph is a geographic area (clustered area) and each edge, e_{ij}, represents the trips from the i^{th} area to the j^{th} area.

Each element of the underlying adjacency matrix (A), a_{ij}, is one if and only if there is an edge between the i^{th} and j^{th} areas in $\boldsymbol{E}_t^d$. Each element of the weight matrix, w_{ij}, denotes the number of trips from v_i to v_j. As stated before, we look at 6-hour time ranges: (1) 1am-6am, (2) 7am-12pm, (3) 1pm-6pm, and (4) 7pm-12am. Hence, per day, d, depending on the nature of the day (weekend/weekday), we have four different graphs: $G_1^d, G_2^d, G_3^d, G_4^d$.

4.1.1 Local and Egonet Features

In order to extract signatures of the traffic network of taxis, for each node we extract the following local and egonet features:

- Degree In: The number of edges going into the node,
- Degree Out: The number of edges going out from the node,
- Weight In: The total weights of edges going into the node,
- Weight Out: The total weights of edges going out from the node,
- $DistIn_i$: The average geographical distance traveled to reach the node,
- $DistOut_i$: The average geographical distance traveled from the node,
- N_i: Number of nodes in egonet i,
- E_i: Number of edges in egonet i,
- $\mathcal{W}_i$: Total weight of egonet i,
- λ_i: Principal eigenvalue of the weighted adjacency matrix of egonet i, and
- Clustering coefficient: the ratio of links that exists in egonet i divided by maximum possible number of links that could exist.

Egonet features are helpful in characterizing the topology and relationships between nodes in the graph. Akoglu et al. [1] find anomalies in a static graph using power law representations. According to their definition, anomalous nodes can have different types w.r.t their egonet features (near-clique, stars, heavy vicinities, and dominant heavy links). We deploy this approach to identify interesting and unusual locations that have such behaviors in their egonets. More details are provided in Section 5.

4.2 Average Probabilistic Flow Graph

One might use a simple approach to look at locations individually to find the profile of each location and also to find occurrence of outliers. Fig. 5(b) shows such an example where traffic bursts (black dots) and absenteeism (red dots) occurred in terms of deviation in the number of pick-ups in one location (vicinity of Times Square – 8th Ave and 40th St). The shaded pink area shows the normal range of variations in the number of pick-ups ($\mu \pm 3\sigma$). Any values outside this range can be labeled as an outlier. We refer to the values higher (or lower) than the normal range as bursts (respectively, absenteeism). However with this technique the relationships between different locations cannot be recovered.

The average graph of traffic flow will be an indicator of normal pattern of locations throughout the year. We calculate this graph using a probabilistic approach. Also this graph can be considered as a baseline model to identify anomalies at different times/days of the year.

Definition 2. Average Probabilistic Flow (APF) Graph: A graph of taxi flows at time t averaged over a set of days, S, $|S| \leq 365$ is $\boldsymbol{G}_t^S(\boldsymbol{V}, \boldsymbol{E}_t^S)$, where $\boldsymbol{V} = \{v_1, v_2, ..v_n\}$ is a set of nodes. The edges $\boldsymbol{E}_t^S$ represent the probability of trips between nodes.

Since we aim to suppress the effect of specific events—where traffic flow happens during a few times with high ratio—the elements of the probabilistic adjacency matrix (A^S) are calculated as follows:

$$a_{ij}^S = \frac{1}{|S|} \sum_{d=1\,, d \in S}^{365} a_{ij}^d. \quad (1)$$

This equals the average number of days that have at least one trip from v_i to v_j and can be interpreted as the empirical probability of having an edge from i^{th} to j^{th} areas. Similarly, the elements of the weight matrix are calculated as follows:

$$w_{ij}^S = \frac{1}{|S|} \sum_{d=1\,, d \in S}^{365} w_{ij}^d. \quad (2)$$

This equals the total number of trips from v_i to v_j in S divided by the number of days ($|S|$) which can be interpreted as the expected number of daily trips from v_i to v_j. Since city mobility patterns differ between weekdays and weekends, we perform separate experiments on each category of days. It should be noted that for weekends $|S| = 104$ and for weekdays $|S| = 261$.

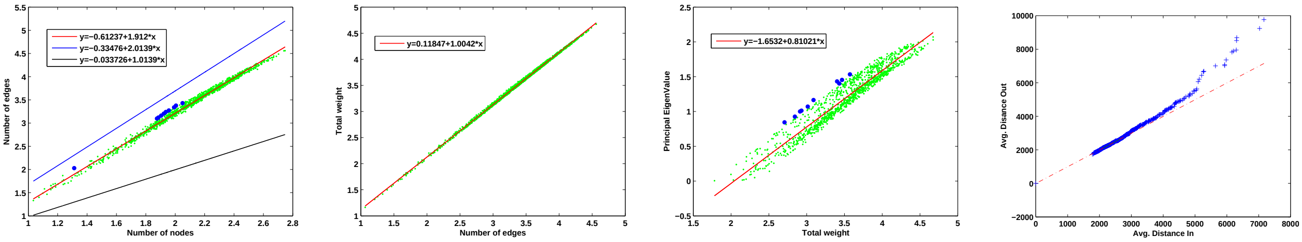
4.2.1 Calculating Local and Egonet Features

In a regular graph, $\boldsymbol{G}_t^d(\boldsymbol{V}, \boldsymbol{E}_t^d)$, we say v_j is in the egonet of v_i, if $e_{ij} \in \boldsymbol{E}_t^d$ or $e_{ji} \in \boldsymbol{E}_t^d$. However in the APF graph, we have to describe features in probabilistic terms. Hence, the probability of the existence of node v_j in egonet i depends on the probability of existence of e_{ij} and e_{ji}. Assuming that the presence of these two edges are independent, the following equation is used to calculate the probability of having node v_j in egonet i:

$$\begin{aligned} P_j^i &= Prob(v_j \in Egonet_i) \\ &= Prob(e_{ij} \in \mathbf{E}_t^d \cup e_{ji} \in \mathbf{E}_t^d) = a_{ij}^S + a_{ji}^S - a_{ij}^S a_{ji}^S. \quad (3) \end{aligned}$$

Recall that a_{ij}^S is the probability that an edge exists from v_i to v_j, i.e. $a_{ij}^S = Prob(e_{ij} \in \mathbf{E}_t^d)$. Then, the expected number of nodes in the egonet i can be determined as follows:

$$N_i = 1 + \sum_{k=1, k \neq i}^{n} P_k^i. \quad (4)$$



(a) (b) (c) (d)

Figure 6: Illustration of relationships of features of the APF graph for an evening period (1pm-6pm) during weekdays. (a) E_i vs N_i, (b) E_i vs. $\mathcal{W}$, (c) $\mathcal{W}$ vs. λ_i, (d) Q-Q plot of $DistIn$ and $DistOut$. Green dots in figures (a) to (c) are nodes and blue dots represent outliers.

Note that the i^{th} node always exists in egonet i (with probability of 1).

In order to calculate the expected number of edges in egonet i, we need to consider e_{kj}, if both j^{th} and k^{th} nodes are in egonet i. The probability of existence of both nodes in egonet i is $P_k^i P_j^i$. Hence, the expected number of edges in egonet i is

$$E_i = \sum_{k=1, k\neq i}^{n} (a_{ik}^S + a_{ki}^S) + \sum_{k=1,k\neq i}^{n} \left(\sum_{j=1, j\neq i, j\neq k}^{n} P_k^i P_j^i (a_{kj}^S + a_{jk}^S) \right) \qquad (5)$$

where the first summation is the expected number of edges that are connected to v_i while the second summation represents the expected number of edges that are not connected to v_i. In a similar way, the expected total weight in egonet i is:

$$\mathcal{W}_i = \sum_{k=1,k\neq i}^{n} (w_{ik}^S + w_{ki}^S) + \sum_{k=1,k\neq i}^{n} \left(\sum_{j=1,j\neq i,j\neq k}^{n} P_k^i P_j^i (w_{kj}^S + w_{jk}^S) \right). \qquad (6)$$

The expected principal eigenvalue of egonet i is derived from the weighted adjacency matrix of egonet. Let us assume that $\mathbf{\Omega}_i^S$ is the weight matrix of egonet i, and $\{v_j, v_k\} \in Egonet_i$. Each element of the weight matrix of egonet i is calculated as follows:

$$\Omega_{i_{jk}}^S = \begin{cases} w_{ji}^S & k = i, j \neq i \\ w_{ik}^S & j = i, k \neq i \\ P_k^i P_j^i w_{jk}^S & j \neq i, k \neq i, k \neq j \\ 0 & O.W. \end{cases} \qquad (7)$$

where $P_k^i P_j^i$ is the probability that both of the nodes v_k and v_j are in egonet i.

The average geographical distance originating from v_i, $DistOut_i$, and the average geographical distance going into v_i, $DistIn_i$, are calculated as follows:

$$DistOut_i = \frac{1}{\sum_{j=1}^n a_{ij}} \sum_{j=1}^{n} a_{ij} dist(i,j),$$

$$DistIn_i = \frac{1}{\sum_{j=1}^n a_{ji}} \sum_{j=1}^{n} a_{ji} dist(j,i), \qquad (8)$$

where $dist(j,i)$ is the distance between v_i and v_j, and $dist(j,i) = dist(i,j)$. Note that the averages of Eq. 8 are weighted averages where higher weights are given to those edges that have higher probability of existence in the average graph.

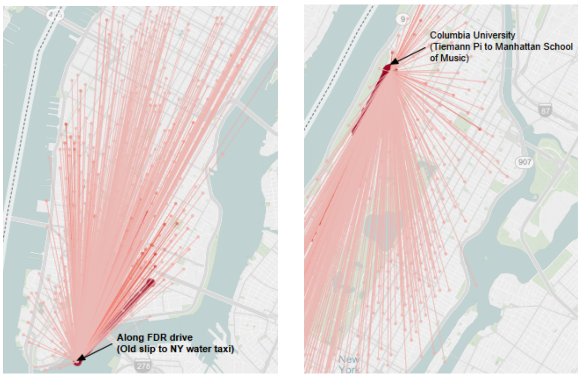
5. BEHAVIOR ANALYSIS USING THE AVERAGE PROBABILISTIC FLOW GRAPH

The egonet features driven from a graph can identify different behavioral patterns of each node [1]. In what follows, we deployed an anomaly detection method based on egonet features [1] on eight APF graphs (four time periods during weekends and four time periods during weekdays) to find locations of interest and investigate their behavior. Since we perform our experiment on an average graph of transportation over a year, outliers detected using this method do not necessarily imply anomalous locations. In fact, this method reveals a set of locations with uncommon features that made them different from the rest of locations. Similar to [1], we analyze the following three pairs of features:

(1) E_i vs N_i: Comparing the number of edges with the number of nodes in each egonet is helpful in detecting near-cliques and stars. According to [1], the number of nodes and number of edges of egonets follow a power law ($E_i \propto N_i^\alpha, 1 \leq \alpha \leq 2$) where in our experiments for APF graphs, α ranges between 1.716 and 1.94. This range of variation for α indicates that most of the nodes have a near-clique pattern. The logarithmic scale for one of the APF graphs (1pm-6pm in weekdays) is shown in Fig. 6(a). The red line shows the least square error fit on data. Also blue and black lines have the slope of 2 (cliques) and 1 (stars), respectively.

(2) $\mathcal{W}_i$ vs E_i: Comparing the total weight with the number of edges in each egonet is helpful in detecting heavy vicinities. The total weight and number of edges follow a power law ($\mathcal{W}_i \propto E_i^\beta, \beta \geq 1$) [1]. In our experiments for APF graphs, β ranged upto 1.023 which reveals that no heavy vicinity node is observed in the traffic flow graph. As Fig. 6(b) depicts, all nodes have similar behavior and no particular node deviates from the fitting line.

(3) λ_i vs $\mathcal{W}_i$: Comparing the principal eigenvalue of weighted adjacency matrix with total weight is helpful in detecting dominant pairs (strongly connected pair of nodes). According to [1], the relationship between these two features follows a power law ($\lambda_i \propto \mathcal{W}_i^\nu, 0.5 \leq \nu \leq 1$) where smaller values of ν indicate uniform distribution of weights while larger values indicate the existence of dominant edges in egonet. In our experiment with APF graphs, ν ranged from 0.766 to 0.906 which means most of the nodes have dominant pairs in



(a) (b)

Figure 7: Illustration of a dominant pair (dark red link) in two locations: (a) Along FDR drive during 1am-6am on weekends, (b) Columbia University during 1pm-12am on weekends.

their egonets. Fig. 6(c) shows an example where most of the nodes have near-heavy edges (ν is close to one). Blue nodes in this figure shows egonets that deviate from the fitting line.

5.1 Locations of Interest

Typically, nodes that deviate from the fitting line are considered as outliers. As we stated before, because we are only studying APF graphs the outliers found by this method should be considered as locations of interest since their patterns are unique (compared to the rest of Manhattan). In Fig. 6(a) and Fig. 6(c), blue circles show the top ten nodes with highest outlier score. Similar to [1], the outlier score for anomaly detection is calculated as a summation of normalized local outlier factor (LOF) and d_f, where d_f is a distance to the fitting line ($y = Cx^\theta$) which is calculated as follows:

$$d_f(i) = \frac{max(y_i, Cx_i^\theta)}{min(y_i, Cx_i^\theta)} * log(|\frac{(y_i - Cx_i^\theta)}{y_i}| + 1).$$

Note that d_f represents the distance between normal behavior (Cx^θ) and the observed value(y) in a normalized logarithmic scale. Higher deviations from the normal behavior (Cx^θ) result in larger values of d_f.

5.2 Discussion on Interesting Points

In what follows, we mention a few samples of discovered interesting locations that either have high feature values or that deviate from the fitting line. It is interesting to know that most of the top attractions of NYC such as the Empire State Building, Rockefeller Center, and the Metropolitan museum of art are *not* included in our list. This suggests that people perhaps use other types of transportation (e.g. subway) to travel to these attractions, or they chose nearby locations as their pick-up and drop-off points.

Fig. 6(d) illustrates Q-Q plots of geographical distances (In vs. Out) in a sample APF graph. As this figure depicts, the distribution of In and Out distances are the same. This is also true for the In and Out degrees of nodes (Fig. 8).

Some locations such as the New York Presbyterian Hospital (covering Fort Washington Ave, from W 161st St to W 173rd St) have high geographical distance to their neighborhoods, indicating that trips to/from this hospital are longer compared to others. High Bridge Park and Claremont Park

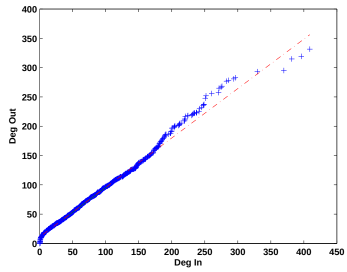


Figure 8: Q-Q plot of Degree-In and Degree-Out of APF graph for Evening(1pm-6pm) of weekdays.

have similar behavior.

Two examples of discovered dominant pairs are shown in Fig. 7. The dark red colored links indicate dominant edges.

The area between Avenue C, E 5th St, and E 3rd St has a near-clique pattern during 7pm-12am. On the other hand, MalcolmX Blvd from W 137th St to W 147th St, during 7am-12pm on weekends has a star pattern.

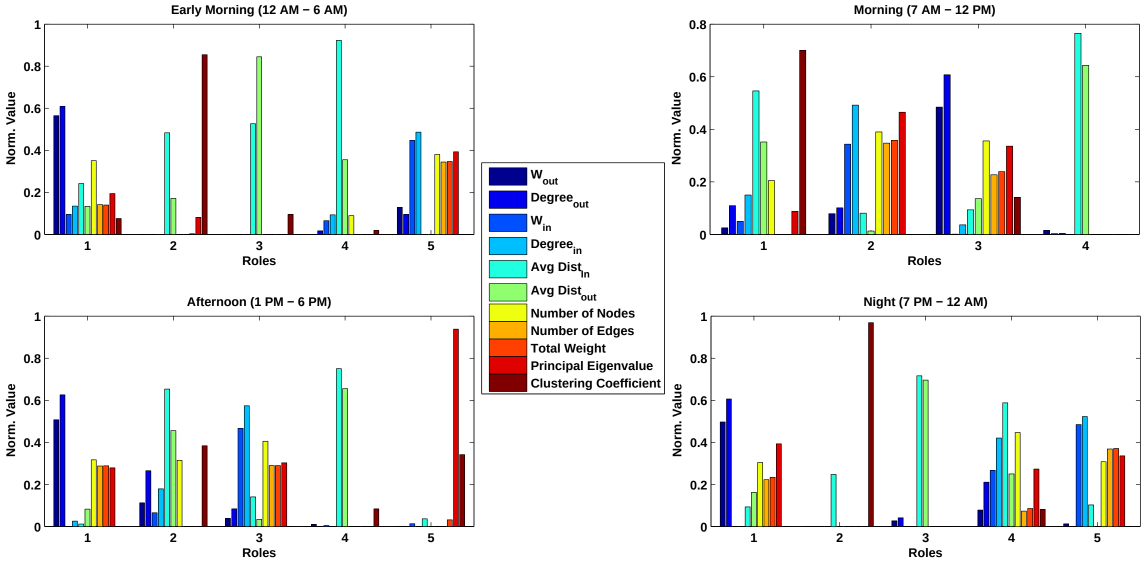
The egonet features of the following locations have high value which made them hotspots: Penn Station (7th Ave and W 31st St) during 7am-6pm on weekends has a high number of nodes, edges, weights, and large eigenvalues. The area covering Penn station (SE), US post office, and intersection of 8th Ave and W 31st St has a high number of incoming and low outgoing trips from 7am-12am. The area covering Chelsea market and Google HQ, in 1am-6am, has a high number of nodes, edges, total weights, and large eigenvalue. The area in Central park (SW), Columbus Circle, 7th Ave, 55th St, and 8th Ave from 7am-12am in weekdays has a high number of nodes, edges, weight, and large eigenvalue.

6. ROLE EXTRACTION

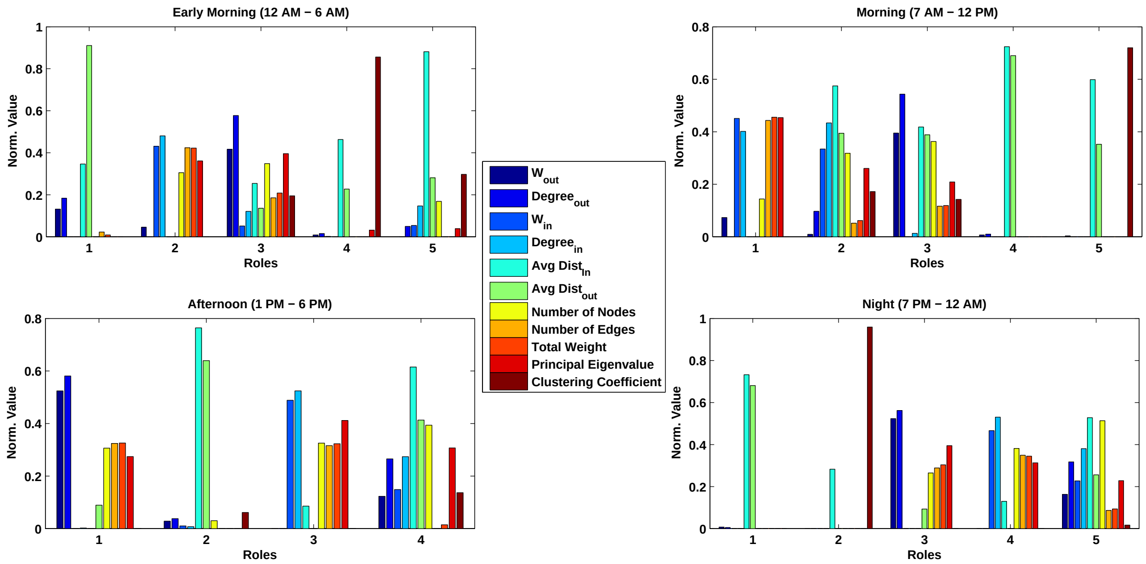
Role extraction (RolX) is a non-parametric, scalable and efficient approach that finds similar structural behaviors and patterns in a graph [10, 21, 22]. The three main steps of this method are feature extraction, feature grouping, and model selection. First, features (global, local, egonet) for each node must be extracted to create a $n \times f$ node-feature matrix X. The next step is to generate a rank r approximation of X using non-negative matrix factorization (NMF). NMF is used to simplify interpretation of roles and memberships by creating non-negative low rank matrices (H_t^S and F_t^S) as follows:

$$H_t^S, F_t^S = \arg\min_{H,F} \frac{1}{2}||X_t^S - HF||_F^2, \; s.t. \, H \geq 0, \; F \geq 0,$$

where X_t^S is the node-feature matrix for the APF graph G_t^S and $||.||_F$ is Frobenius norm. Membership of a node to each role can be estimated using the rows of $H_{n\times r}$ while columns of $F_{r\times f}$ are used to determine the relationship between the role membership and feature values. Since NMF generally results in sparse representations of the original matrix, it is a better candidate for role extraction compared to other factorization methods. The third step is to select the number of roles, r, using the minimum description length (MDL) criterion to compress X. In other words, the objective is to minimize the description length L which is equal to the summation of the coding cost and the cost of model description. MDL selects the number of behavioral roles, r, such that



(a)



(b)

Figure 9: Extracted roles at different time ranges for (a) weekdays, and (b) weekends.

the model complexity (number of bits) and model errors are balanced:

$$L = r(n+f) + (-\frac{1}{2\sigma^2}||X_t^S - H_t^S F_t^S||_F^2),$$

where σ^2 is variance of X_t^S. Details can be found in [10].

The number of detected roles as well as the normalized feature values at each time of weekdays and weekends are shown in Fig. 9. As these figures illustrate, some features were significant in defining the extracted roles such as degrees, weights, and geographical distances.

6.1 Spatio-temporal anomaly detection

The total number of trips per day is shown in Fig. 10. As this figure depicts, the number of trips decreased significantly in specific days with most of them being holidays.

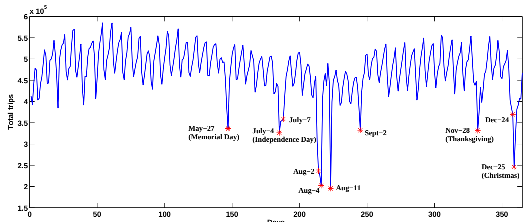


Figure 10: Total trips traveled at each day. Days with high variations represent anomalies.

This might be due to the decrease in the number of available taxi drivers rather than decrease in demand (the present analysis and data availability cannot make this distinction). We use the extracted roles from the APF graph to detect

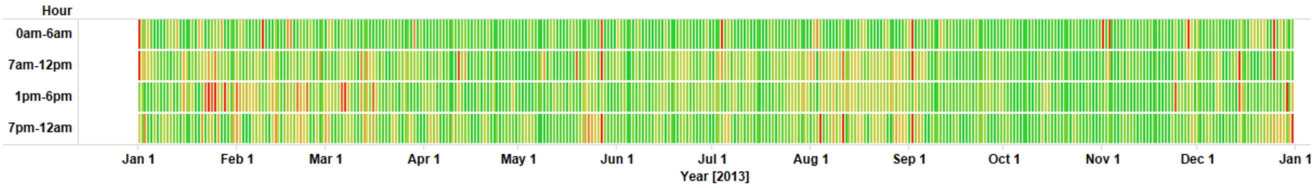


Figure 11: Outliers for different times of a day during a year. Red colors represents higher score of outliers.

anomalies during the whole year. Several techniques have been proposed to detect changes in dynamic networks. As an example, [23] defined an event as a subset of nodes in the network that are close to each other and have high activity levels. However, Rossi et al. [21] track node memberships over time (temporal dependencies of roles and nodes) to discover anomalies. Network dynamics (structural patterns in network over time) can be analyzed using this method. In another work, Rossi et. al [22] proposed dynamic behavioral mixed-membership model to capture roles of the nodes. They identify dynamic patterns in node behaviors and then using prediction on future structural changes in the node, they identify unusual changes in behavior transitions.

In this paper, we use roles extracted from APF graphs and compare the roles with the profile of each day to discover graphs with high variations compared to the average behavior. Therefore, for each graph, G_t^d, we calculate the node-feature matrix X_t^d. Let us assume that based on the APF graph, the role R_i has been assigned to the i^{th} node (i.e. $R_i = \arg\min_k(dist(F^S_{t_{k,:}}, X^d_{t_{i,:}})))$ †. Then we calculate the total distance of graph to the assigned roles in feature space as follows:

$$\Delta_t^d = \sum_{i=1}^{n} ||X^d_{t_i} - F^S_{t_{R_i}}||_2,$$

where n is total number of nodes.

Since we are seeking graphs with high variations, we extract graphs with variations outside the normal range. For this purpose, we calculate the following average and standard deviations over all days at time t:

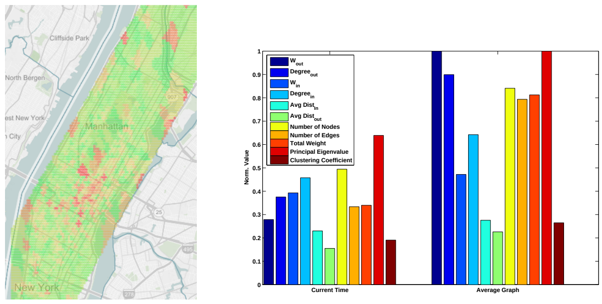
$$\mu_t = \frac{\sum_{d=1,d\in|S|}^{365} \Delta_t^d}{|S|}, \; \sigma_t = \sqrt{\frac{1}{|S|-1}\sum_{d=1,d\in|S|}^{365}(\Delta_t^d - \mu_t)^2},$$

where $|S|$ is number of days. Based on our assumption, an anomaly will occur if the changes in the graph deviates more than a predefined threshold ($\mu \pm 3\sigma$). Fig. 11 shows the variation degree of each part of the day compared to the original assigned roles. Red color shows high variations while green color shows low variations. The result is compatible with Fig. 10. Table 6.1 illustrates the amount of variations for major holidays and cultural events. As an example, the last day of the year (1pm-12am) and first day of the year (1am-12pm) are the ones that have medium to high variations. This result is helpful to understand when the normal behavior of movements in terms of taxi trips changes significantly. As an example, we looked at what happens at each location during Labor day (7am-12pm). The variation at each location is show in Fig. 12(a). Fig. 12(b) shows the difference of features of APF graph (assigned role) vs. that on Labor day for one specific location. The results are helpful for decision-makers and traffic management.

$^\dagger X^d_{t_{i,:}}$ is the i^{th} row of matrix $X^d_{t_{i,:}}$

Table 1: Federal, Religious, and Cultural Holidays with their deviation degrees

Name	1-6	7-12	1-6	7-12
New Year's Day (1/1)	7.1σ	3.5σ	-.7σ	1.6σ
Inauguration Day (1/20)	1.7σ	-1.1σ	.σ	.1σ
M.LutherKing Day (1/21)	1.6σ	1.2σ	.7σ	2.1σ
Groundhog day (2/2)	1.3σ	.7σ	1.8σ	1.4σ
Chinese New Year (2/10)	.3σ	.3σ	.7σ	-.5σ
Lincoln BD (2/12)	-.2σ	1.5σ	1.8σ	1.5σ
Valentine's Day (2/14)	.1σ	.6σ	.8σ	1.2σ
G.Washington BD (2/18)	2.2σ	.9σ	.9σ	1.2σ
Mothers Day (5/12)	-.9σ	-.2σ	-1.σ	-.6σ
Memorial Day (5/27)	2.8σ	4.1σ	-1.σ	3.1σ
Independence Day (7/4)	5.1σ	2.1σ	-.5σ	-.7σ
Eid al-Fitr (8/8)	.2σ	-1.5σ	-1.7σ	-1.9σ
Labor Day (9/2)	5.7σ	3.8σ	-.4σ	4.3σ
Columbus Day (10/14)	-.2σ	-.3σ	-1.4σ	-.2σ
Eid al-Adha (10/15)	-.2σ	-1.1σ	-1.2σ	-.9σ
Halloween (10/31)	.5σ	-.6σ	.6σ	-.3σ
Diwali (11/3)	2.7σ	1.9σ	.3σ	-.6σ
Veterans Day (11/11)	-.3σ	-.4σ	.1σ	-.5σ
Thanksgiving (11/28)	3.6σ	1.6σ	-.8σ	-.6σ
Christmas day (12/25)	6.σ	3.4σ	-.5σ	2.σ
New Year's Eve (12/31)	.7σ	.3σ	1.8σ	5.3σ



(a) (b)

Figure 12: (a) Degree of deviation from assigned roles for Labor day in late morning (7am-12pm) (b) Comparison of assigned role of (Park Ave and Lexington Ave and 62nd St and 60th St) in APF graph and features of Labor Day in late morning.

7. DISCUSSION

In this paper, we applied graph mining approaches to understand the dynamic behavior of taxi trips in a highly populated city. For this purpose, using power-law relationships of egonet features and role extraction using non-negative matrix factorization, we discovered locations of interest as well as outlier days (and locations) at different times. Event prediction methods using the APF graph and utilizing this approach to recommend the placement of new infrastructure are possible directions of future work.

8. REFERENCES

[1] L. Akoglu, M. McGlohon, and C. Faloutsos. Oddball: Spotting anomalies in weighted graphs. In *Proc PAKDD'10*, pages 410–421, 2010.

[2] L. Akoglu, H. Tong, and D. Koutra. Graph-based anomaly detection and description: A survey. *Data Mining and Knowledge Discovery*, 29(3):626–688, May 2015.

[3] M. Z. Austwick, O. OÃćâĆňâĎćBrien, E. Strano, and M. Viana. The structure of spatial networks and communities in bicycle sharing systems. *PLoS ONE*, 8(9):e74685, September 2013.

[4] Z. Cao et al. Analyzing the composition of cities using spatial clustering. In *Proc UrbComp'13*, 2013.

[5] P. S. Castro, D. Zhang, C. Chen, S. Li, and G. Pan. From taxi gps traces to social and community dynamics: A survey. *ACM Comput. Surv.*, 46(2):17:1–17:34, December 2013.

[6] S. Chawla, Y. Zheng, and J. Hu. Inferring the root cause in road traffic anomalies. In *Proc ICDM'12*, pages 141–150, 2012.

[7] T. Cheng, J. Wang, J. Haworth, B. Heydecker, and A. Chow. A dynamic spatial weight matrix and localized space-time autoregressive integrated moving average for network modeling. *Geographical Analysis*, 46:75–97, January 2014.

[8] C. Etienne and O. Latifa. Model-based count series clustering for bike sharing system usage mining. *ACM Trans. Intell. Syst. Technol.*, 5(3):39:1–39:21, July 2014.

[9] Y. Han and F. Moutarde. Analysis of large-scale traffic dynamics in an urban transportation network using non-negative tensor factorization. *International Journal of Intelligent Transportation Systems Research*, pages 1–14, August 2014.

[10] K. Henderson et al. Rolx: Role extraction and mining in large graphs. In *Proc KDD'12*, pages 1231–1239, 2012.

[11] C. Huang and X. Wu. Discovering road segment-based outliers in urban traffic network. In *Globecom 2013 Workshop - Vehicular Network Evolution*, pages 1350 – 1354, 2013.

[12] M. Jiang, P. Cui, A. Beutel, C. Faloutsos, and S. Yang. Catchsync : Catching synchronized behavior in large directed graphs. In *Proc KDD'14*, pages 941–950, 2014.

[13] S. Jiang, J. Ferreira, Jr., and M. C. Gonzalez. Discovering urban spatial-temporal structure from human activity patterns. In *Proc UrbComp'12*, pages 95–102, 2012.

[14] D. Lian et al. Geomf: Joint geographical modeling and matrix factorization for point-of-interest recommendation. In *Proc KDD'14*, pages 831–840, 2014.

[15] W. Liu, Y. Zheng, S. Chawla, J. Yuan, and X. Xie. Discovering spatio-temporal causal interactions in traffic data streams. In *Proc KDD'11*, pages 1010–1018, 2011.

[16] Y. Liu et al. Uncovering patterns of inter-urban trip and spatial interaction from social media check-in data. *PLoS ONE*, 9(1):e86026, January 2014.

[17] X. Min et al. Short-term traffic flow forecasting of urban network based on dynamic starima model. In *IEEE ITCS'09*, pages 1–6, 2009.

[18] M. Momtazpour, P. Butler, M. S. Hossain, M. C. Bozchalui, R. Sharma, and N. Ramakrishnan. Charging and storage infrastructure design for electric vehicles. *ACM Trans. Intell. Syst. Technol.*, 5(3):1–27, September 2014.

[19] S. Ranshous, S. Shen, D. Koutra, C. Faloutsos, and N. F. Samatova. Anomaly detection in dynamic networks: a survey. *Wiley Interdisciplinary Reviews: Computational Statistics*, 7(3):223–247, 2015.

[20] S. Rayana and L. Akoglu. An ensemble approach for event detection and characterization in dynamic graphs. In *ODD'14*, 2014.

[21] R. Rossi and B. Gallagher. Role-dynamics: Fast mining of large dynamic networks. In *Proc WWW'12*, pages 997–1006, 2012.

[22] R. A. Rossi and B. Gallagher. Modeling dynamic behavior in large evolving graphs. In *Proc WSDM'13*, pages 667–676, 2013.

[23] P. Rozenshtein, A. Anagnostopoulos, A. Gionis, and N. Tatti. Event detection in activity networks. In *Proc KDD'14*, pages 1176–1185, 2014.

[24] I. Shafer, K. Ren, V. N. Boddeti, Y. Abe, G. R. Ganger, and C. Faloutsos. Rainmon: An integrated approach to mining bursty timeseries monitoring data. In *Proc KDD'12*, pages 1158–1166, 2012.

[25] D. Singhvi, S. Singhvi, P. Frazier, S. Henderson, E. Mahony, D. Shmoys, and D. Woodard. Predicting bike usage for new york city's bike sharing system. In *AAAI Workshops*, 2015.

[26] H. Tong and C.-Y. Lin. Non-negative residual matrix factorization with application to graph anomaly detection. In *Proc SDM'11*, pages 143–153, 2011.

[27] Y. Wang, Y. Zheng, and Y. Xue. Travel time estimation of a path using sparse trajectories. In *Proc KDD'14*, pages 25–34, 2014.

[28] Y. Wu, F. Chen, C.-T. Lu, and B. Smith. Traffic flow prediction for urban network using spatio-temporal random effects model. In *Transportation Research Board (TRB) 91st Annual Meeting*, 2012.

[29] M. Xu, J. Wu, Y. Du, H. Wang, G. Qi, K. Hu, and Y. Xiao. Discovery of important crossroads in road network using massive taxi trajectories. In *UrbComp'14*, 2014.

[30] J. Yang and J. Leskovec. Patterns of temporal variation in online media. In *Proc WSDM'11*, 2011.

[31] J. Yuan, Y. Zheng, X. Xie, and G. Sun. T-drive: Enhancing driving directions with taxi drivers' intelligence. *Knowledge and Data Engineering, IEEE Transactions on*, 25(1):220–232, January 2013.

[32] Y. Zheng et al. Urban computing: Concepts, methodologies, and applications. *ACM Trans. Intell. Syst. Technol.*, 5(3):38:1–38:55, September 2014.

[33] Y. Zheng, T. Liu, Y. Wang, Y. Zhu, Y. Liu, and E. Chang. Diagnosing new york city's noises with ubiquitous data. In *Proc UbiComp'14*, pages 715–725, 2014.

[34] X. Zhu and D. Guo. Mapping large spatial flow data with hierarchical clustering. *Transaction in GIS*, 18(3):421–435, June 2014.