Intersection Congestion Pattern and Traffic Bottleneck Identification in Cities

Course: CSE 6242

Semester: 2019 Fall

Team No.: Team 31

Members: Chenjun Tang

Danrong Zhang

Jingyu Li

Linwei Gao

Xiangyi Li

Xinze Wang

1. Introduction

Traffic congestion is a severe issue in every big city around the world that widely influences people' life quality, urban management, economic efficiency and air quality. Exploring congestion patterns, mapping the spatial distribution of congested nodes and identifying the traffic bottlenecks can lead to targeted optimization. Based on the data of intersection congestion in four major American cities (Atlanta, Boston, Chicago and Philadelphia), our project is aimed at:

1) Splitting the congested intersections into different categories, so as to describe different congestion patterns at the intersection.
2) Exploring the spatial characters of all the congested intersections in a city so that we can identify the main traffic bottlenecks.

2. Literature Survey

Research on traffic congestion has a long history on both theoretical and practical levels, ranging from traffic data collecting and processing [1], identification of traffic events and patterns [2], exploration of congestion mechanism [3] to traffic data visualization [4], and traffic estimation and forecasting [5]. In all these researches, probe vehicle (e.g. taxis) data is the main method for collecting traffic information. In our project, we also used an open source data of probe vehicles from Kaggle, which mainly describes the congestion state at different intersections in different time of a day.

2.1. Congestion Patterns at Intersection

On the intersection level, our project plans to explore the congestion patterns at each intersection and split intersections with different patterns into different categories. Previous researches mainly described the road states and congestion patterns on an abstract road segment level [1, 6, 7, 16], without distinguishing intersections, roads and other traffic entities explicitly. However, Brunauer etc. [2] proposed that distinguishing road entities can be more expressive compared with abstract road segments. In their research, they defined delay rate of the probe cars by comparing velocity during normal time and congested time, and then used self-organizing maps to recognize and visualize spatio-temporal traffic patterns at intersections [2].

Our project will follow Brunauer's perspective to describe the congestion patterns at realistic intersections. In the meantime, we will further to cluster intersections with similar states into one category to get more abstract understanding of intersection congestion.

2.2. Spatial Clustering in Traffic Data Analysis

On the city level, we will explore the spatial characters of congested intersection to identify the bottleneck in city-wide range. Clustering is a widely used technique in traffic research. Traditionally, researchers focus on continuous travelling traces and apply different clustering methods like k-means [8], piece-wise linear curves [9], fuzzy fitting [5], density-based clustering[10], self-organizing maps [2] and other related modified algorithm [11, 12] to split data points with different location and velocity, so that road segments with different traffic state and spatial location will be separated. But unlike previous researches which rely on continuous GPS traces, our data are spatially separated spots (intersections). Hence, the traditional clustering methods are not suitable for our project.

AMOEBA (A Multidirectional Optimum Ecotope-Based Algorithm) is another method for spatial clustering, which can effectively find clusters of weighted spatial units and achieve spatial contiguity among spots [13, 14]. AMOEBA is not as sensitive as the traditional clustering methods (e.g. k-means [8] and DBSCAN [10]) to spatial distance. It also supports the identification of irregularly shape of spatial clusters. Sun etc. [15]

applied AMOEBA in a practical research to explore the spatial distribution of bars and restaurants in Phoenix, AZ with different scores (Yelp data), which illustrated that AMOEBA was efficient for clustering spatially separated spots. Besides the above researches, AMOEBA is also frequently used to study spatial characters of different events (e.g. neighborhoods and fertility [17], anthrax transmission [18]).

3. Data Analytics Method and Visualization

3.1. Analytics Approach to Recognize Congestion Patterns at Intersection

To explore the congestion patterns among different intersections, we apply clustering to split intersections with different attributes into various segmentations. Although clustering is an unsupervised technique, we can also combine our own prior theoretical assumptions with unsupervised data-driven process. Based on our dataset, our goal is mainly clustering the intersections based on the congestion time of different routes at the intersection. The data transformation framework and the data set we use in clustering is illustrated in Figure 1. And considering that traffic events are highly temporal based, we split the data into different time scenarios (e.g. Time window and weekend or weekdays).

Beyond traditional K-means method, we use Gaussian Mixture Model (GMM) to conduct clustering. The motivation for using Gaussian Mixture Model is that the distribution of the waiting time looks multi-modal (e.g See Figure 2). Applying unimodal model (e.g K-means) to fit the multi-modal distribution usually generate a poor fit. We assume the multi-modal distribution in our dataset is generated from a mixture of several unimodal distributions so GMM is more adequate for our project. Another reason we try GMM is that although the waiting time for each route (e.g left-turn, right-turn and going-straight) at an intersection may be significantly different, there still exists covariant relationship between them. Compared to K-means, GMM facilitate different types of covariance settings so that even very oblong, stretched-out clusters can be fitted.

Figure 1. Data Set Description and Transformation

Original Data Set → Data Transformation and Aggregation → Data Set for Clustering

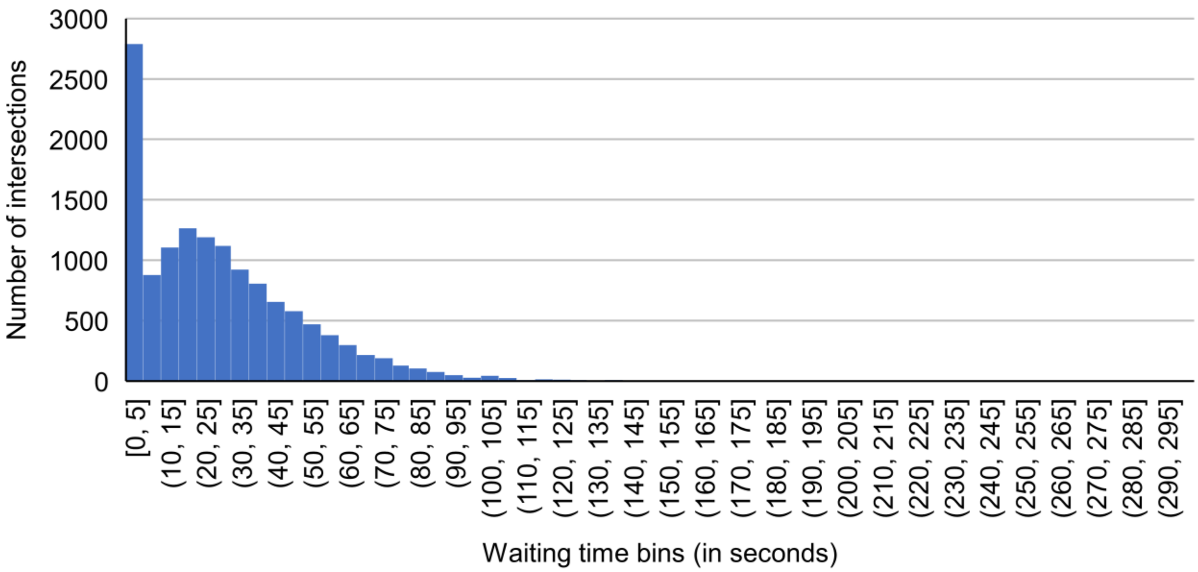
Original Data Set

Variable	Description	Example
InterId	Intersection ID	Philadelphia_244
Latitude	Latitude of intersection	39.93431
Longitude	Longitude of intersection	-75.15081
Hour	One-hour time window in a day	20
Weekend	Weekend or not (weekday)	0
Month	The month that data is collected	11
EntryHeading	Entry direction of the probe vehicle	S
ExitHeading	Exit direction of the probe vehicle	W
TotalTimeStopped_p20	In the corresponding month, all weekends or weekdays and the one-hour time window indicated above, the percentile value of total time stopped collected from all the probe vehicle which drive through the intersection via the specific entry heading and exit heading	8
TotalTimeStopped_p40		16
TotalTimeStopped_p50		18
TotalTimeStopped_p60		21
TotalTimeStopped_p80		36

Data Set for Clustering

Variable	Description
InterId	Intersection ID
Latitude	Latitude of intersection
Longitude	Longitude of intersection
Time Window	Divide 24 hours in a day into several parts, e.g. morning busy, evening busy, midnight etc.
Weekend	Weekend or not (weekday)
Left Turn TotalTimeStopped	Transform entry and exit heading into driving route. And Choose TotalTimeStopped_p80 as our metrics, and aggregate the data on intersection level by using the median number
Right Turn TotalTimeStopped	
Straight TotalTimeStopped	

Figure 2. Intersection Waiting Time Distribution during Weekday Evening Busy Hours



Notes: We take weekday evening busy hours as an example. The distributions during other time slot are similar.

The algorithm concept of GMM is similar with K-means. GMM uses Expectation-Maximization approach, which iterates between the "Expectation" step – to find weights encoding the probability of membership in each cluster for each data points, and the "Maximization" step – update the shape and location of each clusters based on the weights from all data points.

The metrics we use to evaluate the model are Bayesian information criterion (BIC) and Silhouette Score. BIC is defined as:

$$BIC = ln(n) * k - 2 * ln(\hat{L})$$

in which n is the number of data points,

k is number of parameters estimated in the model,

$\hat{L}$ *is the maximum of likelihood.*

In general, adding parameters when fitting the model can increase the likelihood, but overfitting may also happen. BIC balances between the fit of the model and the complexity of the model. The lower value of BIC is, the better the model is. Silhouette Score is used to evaluate the consistency of the clusters of data. It's defined as:

$$SilhoueteeScore = \frac{b - a}{max(a, b)}$$

in which b measures distance between a data point and the nearest cluster

a measures distance between a data point and its own cluster

To calculate the overall Silhouette Score of the whole dataset can evaluate how the data points are close to their cluster and how they are away from other clusters.

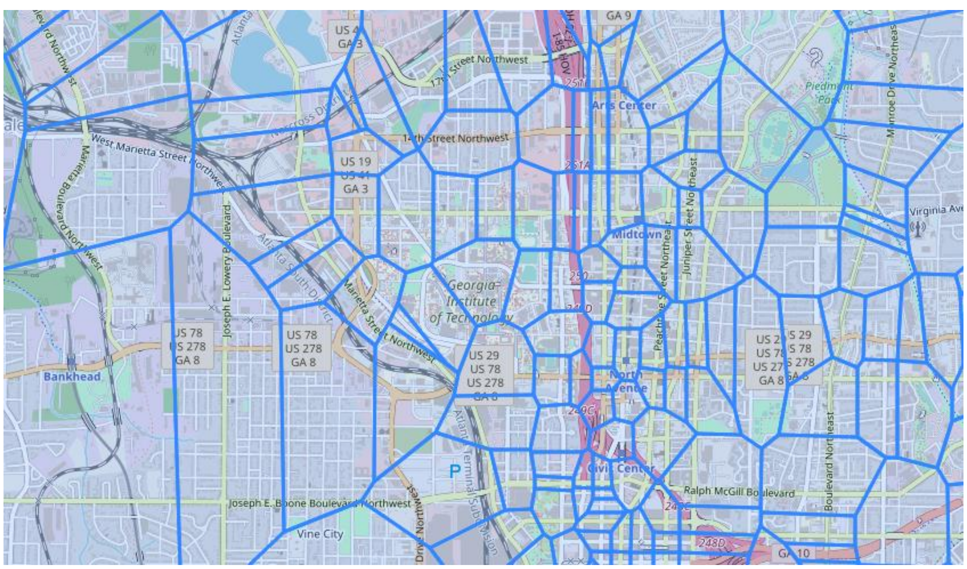
3.2. Analytics Approach for Spatial Clustering and Bottleneck Identification

Our second goal is to group adjacent intersections into spatial clusters, which is accomplished by AMOEBA algorithm [13, 14]. AMOEBA constructs a spatial weights matrix using empirical data that can also simultaneously identify the geometric form of spatial clusters.

The reasons why we apply AMOEBA algorithm in our project lie in two aspects. First of all, AMOEBA is not as sensitive as the traditional clustering methods (e.g. k-means [8] and DBSCAN [10]) to spatial distance. In clustering, the weight of spatial distance will be lower and the similarity on other attributes (e.g. congestion time) among data points can be reflected in the clustering results. Secondly, while traditional clustering techniques generally identify data points in a circular range as one cluster, AMOEBA supports the identification of irregularly shape of spatial clusters [15], which is more suitable based on the project's dataset.

To apply AMOEBA, the spatial units should be spatially contiguous. However, in our project, intersections are represented by points as spatial units. To bridge the gap, we generate Voronoi polygons around intersections. Each intersection's corresponding polygon consists of all points closer to that intersection than to any other. We use python library PySal. The inputs of the algorithm are the coordinates of the intersections, and the outputs include the shape file of polygons and their neighboring relationships.

Figure 3. Voronoi Polygon Example

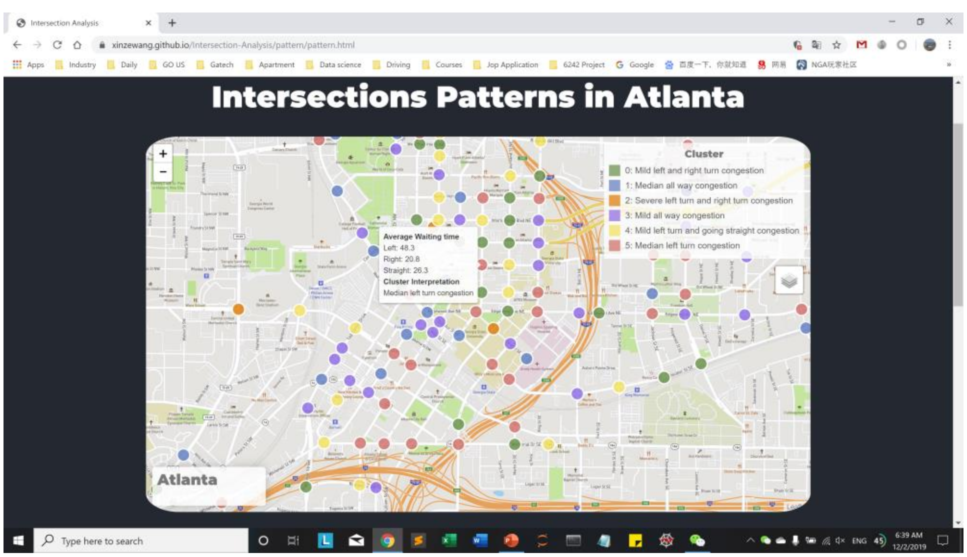


Then we run AMOEBA algorithm to find the spatial clusters among polygons. We use AMOEBA library in R. It searches for spatial association in all specified directions from one or more polygons and reveals all the spatial association that is subsumed in the data. And finally, it divided all the polygons into different contiguous spatial units and assign weight to these units based on their spatial association. The inputs of the AMOEBA algorithm include the attribute of polygons, shape file of polygons and their neighboring relationships.

3.3. Data Visualization Approach

The visualization of our project will be web-based and GIS-based. Users can get a high-level overview of intersections with different congestion patterns and spatial groups of intersections on the map. Our website is built mainly based on HTML and CSS, with responsive web design using Bootstrap framework and interesting symbols from Font Awesome. To make our website more interactive, we used jQuery to manipulate styles, animations and customizations. To design our city map with simplicity and usability, we used open-source JavaScript library Leaflet to make it more user-friendly. Finally, we use git to deploy our website on Heroku application platform.

Figure 4. Visualization User Interface Illustration



4. Data Analytics and Experiment

4.1. Data Preparation

The first step is data cleaning and data transformation. In this process, the main problem we need to deal with is that when we aggregated the original data according to the time windows and driving routes (as shown in Figure 1), there are some missing data because no probe car drove through a certain route at an intersection during a time window. We used MICE method (Multiple Imputation by Chained Equations) to impute the value. The algorithm of MICE is that for each feature with missing values, building a prediction model (e.g. regression) based on other features to predict and impute the missing values. Then this process iterates among all the features for several rounds until the predicted missing values convert to a stable level.

4.2. Gaussian Mixture Model for Recognize Congestion Patterns

4.2.1. Clustering Results Summary

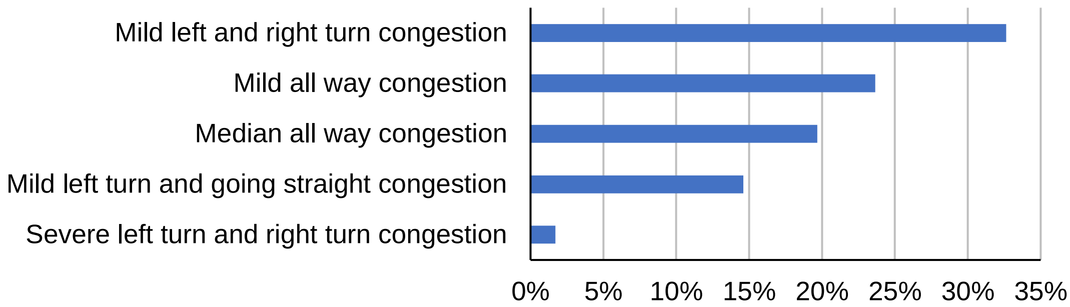
The table below summarize the congestion patterns we recognize from the data via Gaussian Mixture Model. Table 1 illustrate the results for each time window during weekdays and Table 2 illustrate the results during weekends.

From the results, we notice that the congestion patterns are described by information in two dimensions: the degree of congestion (general length of waiting time) and the difference in congestion time among driving routes (different waiting time in left turn, right turn and going straight). For the time slots during weekdays, although the congestion patterns are not identical to each other, there exists some patterns appearing across different time window. The results show that "Mild all way congestion", "Median all way congestion", "Mild left and right turn congestion", "Mild left turn and going straight congestion", and "Severe left turn and right turn congestion" are the typical congestion patterns, in which "Mild left and right turn congestion" intersections appears more widely (See Figure 5). Meanwhile, the number of “Severe left turn and right turn congestion” intersections is small, but they demonstrate a typical congestion pattern that can distinguish them with other intersections.

Table 1. Average Waiting Time of Each Cluster and Interpretation (Weekdays; in seconds)

Time	ID	Average Waiting time			Num. of Intersection	Cluster Interpretation
		Left	Right	Straight		
MorningBusy (07:00-10:00)	0	91.1	102.1	43.2	77	Severe left turn and right turn congestion
	1	30.5	0.0	19.3	504	Mild left turn and going straight congestion
	2	30.0	24.3	17.5	1878	Mild all way congestion
	3	26.2	17.6	0.0	1316	Mild left and right turn congestion
	4	57.1	43.6	37.8	930	Median all way congestion
NormalDay (10:00-16:00)	0	21.3	18.8	0.0	1080	Mild left and right turn congestion
	1	48.1	49.3	32.7	807	Median all way congestion
	2	87.7	64.8	51.3	132	Severe left turn and right turn congestion
	3	22.2	22.2	15.1	1314	Mild all way congestion
	4	33.6	0.0	13.3	652	Mild left turn and going straight congestion
	5	48.3	20.8	26.3	751	Median left turn congestion
EveningBusy (16:00-20:00)	0	24.6	17.9	0.0	1143	Mild left and right turn congestion
	1	55.7	39.3	35.6	1224	Median all way congestion
	2	113.0	55.4	70.6	104	Severe left turn congestion
	3	21.2	32.4	16.0	798	Mild all way congestion
	4	32.2	8.0	17.6	1162	Mild left turn and going straight congestion
	5	70.3	143.1	39.8	48	Severe right turn congestion
NormalNight (20:00-22:00)	0	23.7	16.9	0.0	1419	Mild left and right turn congestion
	1	28.3	23.3	23.2	648	Mild left turn congestion
	2	17.7	22.0	10.0	486	Mild all way congestion
	3	28.6	0.0	17.6	394	Mild left turn and going straight congestion
	4	63.3	23.2	13.3	179	Median left turn congestion
	5	52.3	45.7	35.0	492	Median all way congestion
Midnight (22:00-07:00)	0	27.9	0.0	16.8	498	Mild left turn and going straight congestion
	1	22.6	15.4	0.0	2035	Mild left and right turn congestion
	2	93.6	87.8	50.0	30	Severe left turn and right turn congestion
	3	43.7	37.5	24.9	884	Median all way congestion
	4	23.2	17.5	13.4	827	Mild all way congestion

Figure 5. Average Proportion of Intersections Belonging to A Cluster



As for the situation during weekends, congestion patterns among different time slots are more similar with each other. The intersections are clustered either as 3 clusters with “Mild all way congestion”, “Median all way congestion” and “Mild left and right turn congestion” or as 4 clusters in which “Mild left turn and going straight congestion” is split from the whole set.

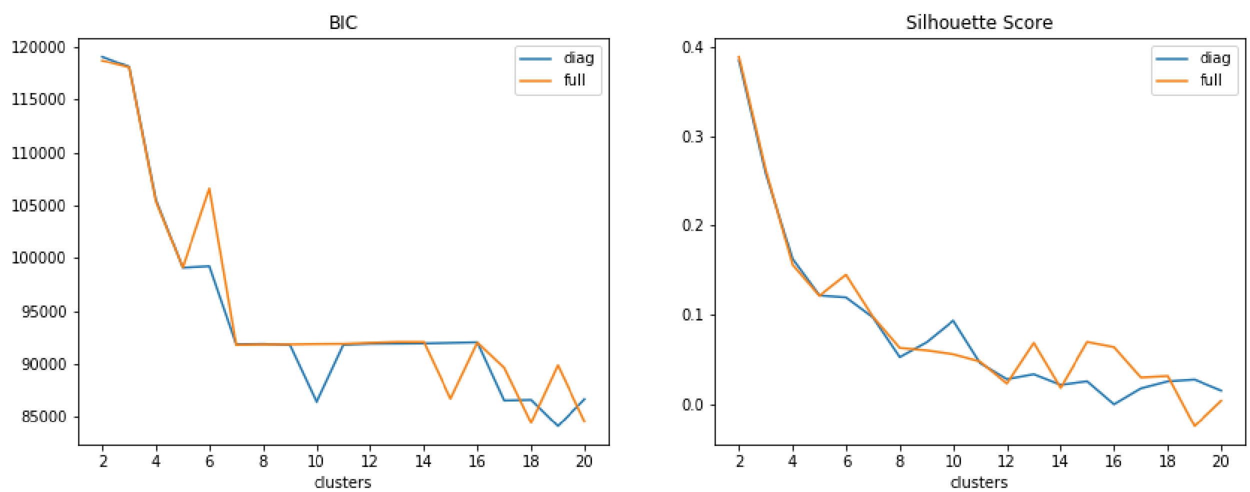
Table 2. Average Waiting Time of Each Cluster and Interpretation (Weekends; in seconds)

Time	ID	Average Waiting time			Num. of Intersection	Cluster Interpretation
		Left	Right	Straight		
MorningBusy (07:00-10:00)	0	30.1	12.9	14.6	896	Mild all way congestion
	1	45.1	42.7	29.8	619	Median all way congestion
	2	99.9	330.3	94.2	3	Extreme situation
	3	24.7	17.1	0.0	1289	Mild left and right turn congestion
NormalDay (10:00-16:00)	0	31.7	0.0	16.8	379	Mild left turn and going straight congestion
	1	31.8	26.6	18.7	1338	Mild all way congestion
	2	63.0	51.3	42.9	401	Median all way congestion
	3	22.4	17.7	0.0	1111	Mild left and right turn congestion
EveningBusy (16:00-20:00)	0	24.5	17.8	0.0	1034	Mild left and right turn congestion
	1	61.3	48.5	42.2	362	Median all way congestion
	2	30.7	20.7	18.0	1449	Mild all way congestion
NormalNight (20:00-22:00)	0	29.9	0.0	16.4	275	Mild left turn and going straight congestion
	1	24.5	23.9	16.1	781	Mild all way congestion
	2	55.8	39.3	33.1	450	Median all way congestion
	3	24.5	17.4	0.0	1000	Mild left and right turn congestion
Midnight (22:00-07:00)	0	23.5	16.7	0.0	1452	Mild left and right turn congestion
	1	56.1	26.1	31.9	428	Median all way congestion
	2	24.5	22.4	15.1	927	Mild all way congestion

4.2.2. GMM Model Evaluation

As an unsupervised and exploratory model, the evaluation process mainly focuses on hyperparameter tuning. In detail, we explore different covariance type settings (e.g. "diag": each component has its own diagonal covariance matrix; "full": each component has its own general covariance matrix) and different number of clusters (e.g. 2 to 20). As mentioned above, the metrics we use to select a relatively better model are Bayesian information criterion (BIC) and Silhouette Score. Taking weekday evening busy hour as an example, the BIC and Silhouette Score plot under different parameter settings are showed below. For some instance, BIC and Silhouette Score may indicate controversial model selection. We determine the parameters by taking both into consideration and making a balance.

Figure 6. Hyperparameter Tuning on Covariance Matrix and Number of Clusters (Weekdays Evening Busy Hours)



Notes: We take weekday evening busy hours as an example. The approach is similar for other time slot conditions.

Based on model evaluation, the parameters for the model in each time slot conditions are illustrated in Table 3.

Table 3. Model Evaluation and Selection

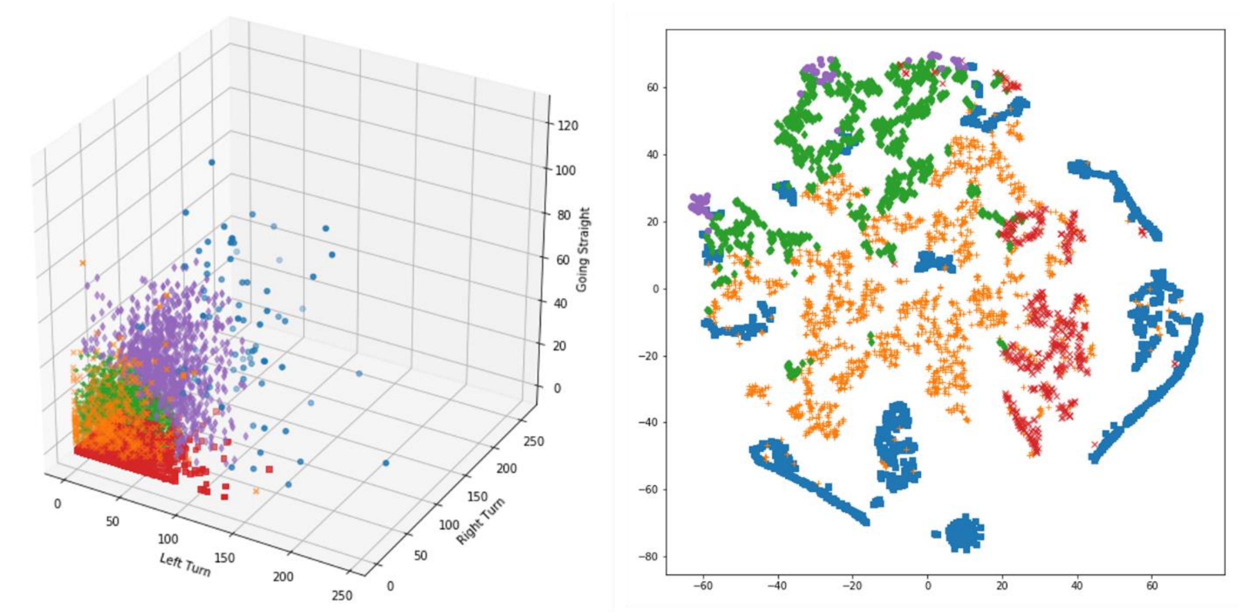
Day	Time Slot	Number of Clusters	Covariance Type
Weekdays	MorningBusy (07:00-10:00)	5	Diag
Weekdays	NormalDay (10:00-16:00)	6	Diag
Weekdays	EveningBusy (16:00-20:00)	6	Diag
Weekdays	NormalNight (20:00-22:00)	6	Diag
Weekdays	Midnight (22:00-07:00)	5	Full
Weekends	MorningBusy (07:00-10:00)	4	Full
Weekends	NormalDay (10:00-16:00)	4	Diag
Weekends	EveningBusy (16:00-20:00)	3	Diag
Weekends	NormalNight (20:00-22:00)	4	Diag
Weekends	Midnight (22:00-07:00)	3	Full

4.2.3. 3-Dimension Clustering Results Visualization

We use waiting time of left-turn, right-turn and going-straight to run clustering model, how to visualize the results is another interesting topic. Besides traditional 3-dimension plot, we also try T-Distributed Stochastic Neighboring Entities (t-SNE) technique to do dimension reduction.

The general algorithm of t-SNE is to “minimize the divergence between distribution that measures pairwise similarities of the input objects and distribution that measures pairwise similarities of the corresponding low-dimensional points in the embedding” [19]. Taking weekdays morning busy hours as example, the 3-dimension plot and 2-dimension plot after t-SNE transformation are presented below.

Figure 7. Clusters Visualization Plot: Left-3D plot and Right-TSNE plot (Weekdays Morning Busy Hours)



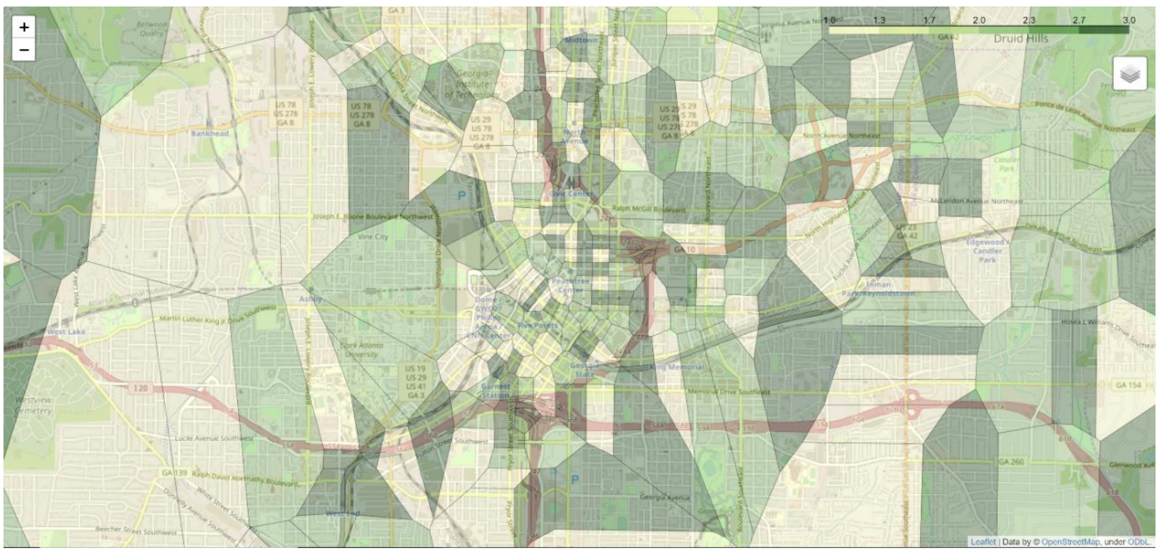
Comparing the two plots, we propose that t-SNE plot is more straight-forward for us to get a high-level impression about how the data points are clustered. Transforming the data from high dimensions to two dimensions results in a degree of losing information, but two-dimension plot is much easier for human to perceive and understand.

4.3. AMOEBA Approach for Spatial Clustering and Bottleneck Identification

AMOEBA is an algorithm for spatially constrained clustering and spatial aggregation. It combines the spatial connection relations among each spot and the attribute of the spot (e.g. waiting time in our project) to split them into different categories. By applying AMOEBA, the clustering results generated not only identify which cluster the spot belongs to, but also represent the relative degree of congestion among each cluster.

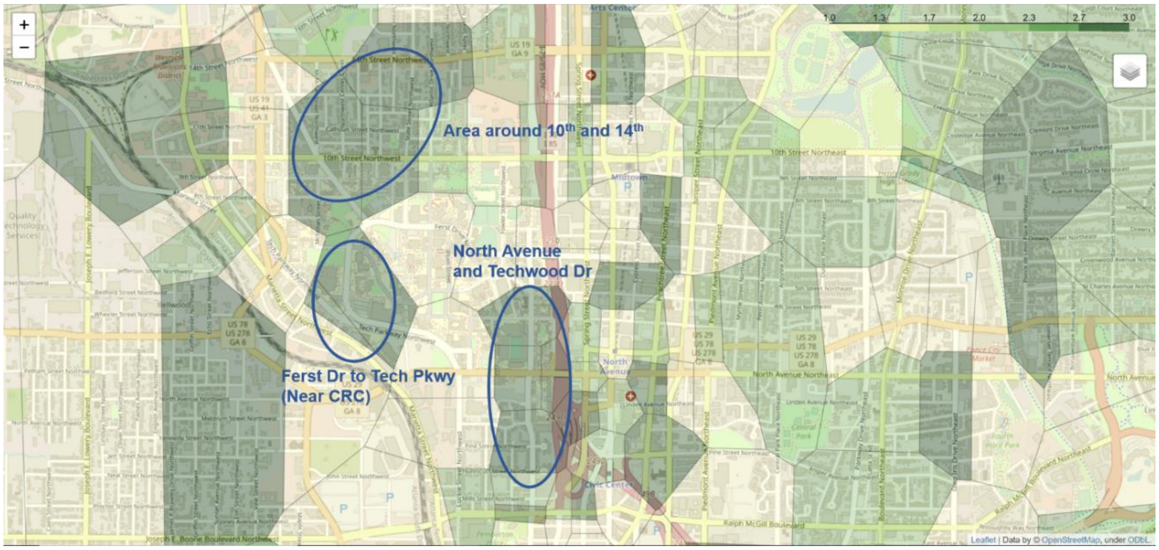
In the project, we run AMOEBA on the sub-dataset of Atlanta and visualize the results via choropleth maps. In Figure 8, each polygon represents an intersection. We notice that the area of one polygon may cover several intersections on the map. The reason is that the dataset only contains the congestion data of one intersection in that area, as probe car for collecting data didn't drive through all intersections. The darkness of color represents the relative degree of congestions: the darker, the severer.

Figure 8. Example of Choropleth Map for Spatial Clustering (Weekdays Morning Busy Hours)



The goal is to identify traffic bottlenecks in the city. We propose that the connected polygons with relatively high degree of congestion are potential bottlenecks. The first set of bottlenecks we can identify from the analysis is the main entrances and exits around Georgia Tech (Figure. 9). Considering that GT campus is a hot area gathering large number of people, the intersections and roads entering and exiting the campus will bear large traffic, especially during morning busy hours and evening busy hours.

Figure 9. Bottlenecks Identified around Georgia Tech



Notes: The choropleth map illustrated is during weekday evening busy hours. The situation is similar during other busy hours.

Another set of bottlenecks identified in the project is the area around Peachtree Center in downtown. The roads are narrow, and road network is complicated, which results in severe congestion.

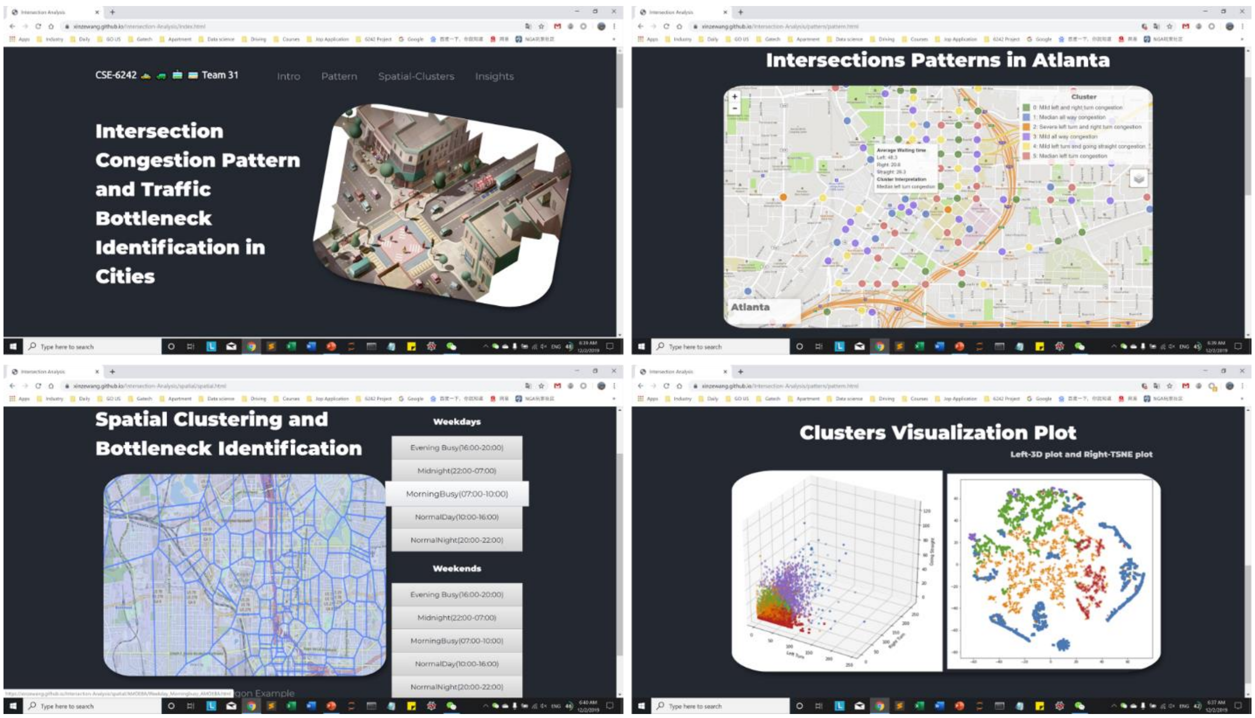
Figure 10. Bottlenecks Identified in Downtown (around Peachtree Center)

Notes: The choropleth map illustrated is during weekday evening busy hours. The situation is similar during other busy hours.

4.4. Web-based and GIS-based Visualization

As proposed above, we build a website to visualize the congestion patterns and spatial clustering of intersections on the map. We provide a demo for users to experience.

Figure 11. Visualization Website Illustration



5. Conclusions and Discussion

5.1. Project Findings Summary

The project is focusing on data analytics and visualization upon the dataset of intersection congestions in four major cities (Chicago, Atlanta, Boston and Philadelphia) in US. The project's goal is: 1) to use clustering techniques to find the congestion patterns among all intersections; 2) to find the spatial clusters of intersections based on the location and congestion situations, so that traffic bottlenecks in the city can be identified.

We propose that the congestion patterns are described by two dimensions: the degree of congestion and the difference in congestion time among driving routes. We found that “Mild all way congestion”, “Median all way congestion”, “Mild left and right turn congestion”, and “Mild left turn and going straight congestion” are some typical congestion patterns widely appears during different time slots.

By using AMOEBA, we also identify two sets of traffic bottlenecks in Atlanta. The first set of bottlenecks is the main entrances and exits around Georgia Tech, including the intersections of Ferst Dr and Tech Pkwy near CRC, the intersections in the area around 10th and 14th street, and area around intersection of North Avenue and Techwood Dr. The second set of bottlenecks is the area around Peachtree Center in downtown.

5.2. Project Innovations

Previous researches rarely focus on the intersection congestions specifically (e.g [1, 12, 10, 16]). We propose that intersections are important elements and entities in the road network. Beyond researching congestion on abstract road segments, we believe it’s meaningful to analyze congestion patterns at different intersections and to explore spatial distribution of congested intersections.

Secondly, to describe intersection congestion patterns based on the congestion time of different routes at a certain intersection is also new compared to previous work. By adopting this perspective, we identify several typical congestion patterns (e.g. “Median all way congestion” and “Mild left and right turn congestion”).

Besides, we identify several set of traffic bottlenecks in Atlanta on the intersection level, which will be potential and useful add-on for the congestion optimization in Atlanta.

5.3. Limitation of the Project

The information of the dataset in our project is not intact enough as our expectation. For example, the congestion states in part of the intersections are not collected. We didn't find other open source data to supplement more information for data analysis in this project. It results in the spatial clustering not being accurate enough, especially for the area out of downtown and midtown.

The web-based and GIS-based visualization and user interface have potential to improve, including ease of use, ease of understanding, loading speed, and degree of aesthetic.

6. Team Member Efforts

All team members have contributed similar amount of efforts in our project.

Reference

[1] Liu, X., & Ban, Y. (2013). Uncovering spatio-temporal cluster patterns using massive floating car data. ISPRS International Journal of Geo-Information, 2(2), 371-384.

[2] Brunauer, R., Schmitzberger, N., & Rehrl, K. (2018, November). Recognizing Spatio-Temporal Traffic Patterns at Intersections Using Self-Organizing Maps. In IWCTS@ SIGSPATIAL (pp. 43-52).

[3] Yue, W., Li, C., & Mao, G. (2018, May). Urban Traffic Bottleneck Identification Based on Congestion Propagation. In 2018 IEEE International Conference on Communications (ICC) (pp. 1-6). IEEE.

[4] Lopez, C., Leclercq, L., Krishnakumari, P., Chiabaut, N., & Van Lint, H. (2017). Revealing the day-to-day regularity of urban congestion patterns with 3D speed maps. Scientific Reports, 7(1), 14029-14039.

[5] Tang, L., Kan, Z., Zhang, X., Yang, X., Huang, F., & Li, Q. (2016). Travel time estimation at intersections based on low-frequency spatial-temporal GPS trajectory big data. Cartography and Geographic Information Science, 43(5), 417-426.

[6] Xu, L., Yue, Y., & Li, Q. (2013). Identifying urban traffic congestion pattern from historical floating car data. Procedia-Social and Behavioral Sciences, 96, 2084-2095.

[7] Lopez, C., Krishnakumari, P., Leclercq, L., Chiabaut, N., & Van Lint, H. (2017). Spatiotemporal partitioning of transportation network using travel time data. Transportation Research Record, 2623(1), 98-107.

[8] Necula, E. (2015). Analyzing traffic patterns on street segments based on GPS data using R. Transportation Research Procedia, 10, 276-285.

[9] Ban, X. (Jeff), Herring, R., Hao, P. and Bayen, A.M. 2009. Delay Pattern Estimation for Signalized Intersections Using Sampled Travel Times. Transportation Research Record. 2130, 1, 109–119.

[10] Liu, Y., Yan, X., Wang, Y., Yang, Z., & Wu, J. (2017). Grid Mapping for Spatial Pattern Analyses of Recurrent Urban Traffic Congestion Based on Taxi GPS Sensing

Data. Sustainability, 9(4), 533.

[11] Nath, Rudra Pratap Deb, et al. "Modified K-means clustering for travel time prediction based on historical traffic data." International Conference on Knowledge-Based and Intelligent Information and Engineering Systems. Springer, Berlin, Heidelberg, 2010.

[12] Rempe, F., Huber, G., & Bogenberger, K. (2016). Spatio-temporal congestion patterns in urban traffic networks. Transportation Research Procedia, 15, 513-524.

[13] Aldstadt J, Getis A. Using AMOEBA to create a spatial weights matrix and identify spatial clusters. Geogr Anal. 2006;38(4):327–43.

[14] Duque JC, Aldstadt J, Velasquez E, Franco JL, Betancourt A. A computationally efficient method for delineating irregularly shaped spatial clusters. J Geogr Syst. 2011;13(4):355–72.

[15] Sun, Y., & Paule, J. D. G. (2017). Spatial analysis of users-generated ratings of yelp venues. Open Geospatial Data, Software and Standards, 2(1), 5.

[16] Ji, Y., Luo, J., & Geroliminis, N. (2014). Empirical observations of congestion propagation and dynamic partitioning with probe data for large-scale systems. Transportation Research Record, 2422(1), 1-11.

[17] Weeks, J. R., Getis, A., Hill, A. G., Agyei-Mensah, S., & Rain, D. (2010). Neighborhoods and fertility in Accra, Ghana: An AMOEBA-based approach. Annals of the Association of American Geographers, 100(3), 558-578.

[18] Barro, A. S., Kracalik, I. T., Malania, L., Tsertsvadze, N., Manvelyan, J., Imnadze, P., & Blackburn, J. K. (2015). Identifying hotspots of human anthrax transmission using three local clustering techniques. Applied Geography, 60, 29-36.

[19] Van der Maaten, L., & Hinton, G. (2008). Visualizing data using t-sne. Journal of machine learning research, 9, 2579-2605.