2022 4th International Conference on Advances in Computing, Communication Control and Networking (ICAC3N) | 978-1-6654-7436-8/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICAC3N56670.2022.10074196

Visualization of Traffic Density on Graph using Trajectory Data

Sahil More *, *Ashwini Matange* §, *Jibi Abraham* †
Dept. of Computer Engineering and Information Technology
COEP Technological University
Pune, India
Email: $^*moresr20.comp@coep.ac.in$, $^§ asm.comp@coep.ac.in$, $^† ja.comp@coep.ac.in$

***Abstract*—Several applications collect users' trajectory data to provide better location-based services. Since the data reveals the mobility of the users as a group and as an individual, publishing this trajectory data may pose a threat to one's privacy. The current trajectory data mining procedures take all the raw information from the trajectory data without considering the users' privacy. This type of data mining can be prone to linkage attacks. This research proposes a novel mechanism that uses OpenStreetMap and its geo-entities recognition feature to extract and cluster the traffic signals to form vertices of a graph. From the trajectory data, vehicle type and the edge in the graph is determined. The vehicle count for each vehicle type on the edge of a graph is iteratively updated and it is visually represented on the respective road edge between two junction vertices. The visualization of the weighted road network graph enables analysts to infer useful information as the road graph itself preserves the statistical information of the road density but hides the individual users' trajectory.**

***Index Terms*—Trajectory data, trajectory data mining, utility, road graph, junction-based trajectory data.**

I. Introduction

Companies that provide Location Based Services(LBS) currently collect user mobility data to offer services based on that data. This data provides many features, including traffic flow, human mobility across regions, the density of vehicles and people across different areas, etc. With this kind of helpful information, the data should be designed to be feasible for every analyst to extract precisely what is required.

First off, the majority of technologies in use today cannot effectively analyze the crucial functions that city roadways play in actual traffic conditions. Traditional grid-based methods handle spatial data in a city's Euclidean space, which does not accurately reflect the network structure of roadways. In geographical and transportation domain, road structures and their network have been studied using graph approaches based on static street networks. However, because they do not use actual traffic data, they cannot accurately portray the transit functions of roadways. Although recent research has been done on using urban trajectory data, it does not provide graphs that depict city-wide road networks or enable interactive analysis of route significance.

Secondly, in order to visually examine their duties, domain users must interactively choose city roadways. The use of visualization in the exploratory analysis is desired because it enables users to use their subject expertise and human intelligence. Since street networks in big cities are fairly wide and detailed, interactive visualization is a challenging endeavor due to the magnitude and complexity of the data. The problem is further complicated when dynamic traffic information from enormous trajectory data is included. Effective and fast computational techniques should be strongly linked with clear visuals to enable interactive visual exploration. Because there aren't any visible analytics solutions, domain users are being hindered. To address this problem, this research proposes a novel mechanism to convert raw trajectory data into a junction-based road graph to fulfill this purpose. Trajectories of a single user contain thousands of location points between two junctions. Representation of trajectory in a junction-based road network graph enables minimization of the location points as the trajectory transition from junction to junction is published. This mechanism utilizes the K-means clustering algorithm, distance function like Haversine distance, and a novel density calculation algorithm to produce the required output.

A. Trajectory Data

Human mobility information is collected by several organizations to optimize their location-based services. One form of such data is the trajectory data of users. Trajectory data is the geographic location points of users arranged in chronological order with respect to time. The mobility of a variety of moving things, including humans, cars, and animals, is represented by trajectory data. Over the past years, a wide range of strategies has been presented for processing, maintaining, and mining trajectory data [13].

The trajectory data don't need to be ready to mine when gathered; sometimes, data is structured correctly with aspects like time and geographic location apparent and in good continuity. This type of data doesn't require a lot of effort to process. An example of this type of data is collected from GPS-enabled devices, including each location point's latitude, longitude, and timestamp. GPS data is commonly used in trajectory data mining. On the other hand, some data collected using crowdsensing technology have little continuity in location coordinates, resulting in less accuracy while processing the trajectory data [16].

ISBN:978-1-6654-7436-8/22/$31.00 ©2022 IEEE

B. Utility of Trajectory Data

Trajectory data of users can be of many uses including transportation management, which analyses the mobility of users with respect to road traffic to make necessary decisions to make traffic flow more efficient. In pandemic tracking, trajectory data is used to infer how humans move from region to region which helps in tracking how an infection spreads in communities. The main use of trajectory data that we are addressing in this research is urban planning. Trajectory data is mined in urban planning to analyze the movement of people, vehicles, etc to take the precise business decision regarding urban development. For e.g., a construction plot around a road segment with heavy traffic most of the time might not be the right place to plan a residential building structure. Fig. 1. shows the open street map of a residency in Beijing city and Fig. 2. show the location point plotted on that open street map which shows the trajectory of users across that area.

Fig. 1. Open street map of a residential structure.

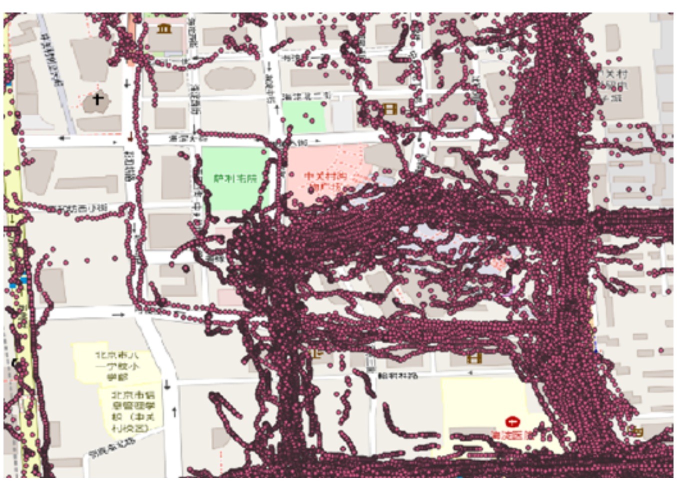


Fig. 2. Location points plotted on the open street map.

II. Literature Review

To mimic geography, Rozenblat et al. developed a project. They examined multilevel spatial networks produced by comples effective systems, like hierarchical structure of cities organized by the amount of authority held over them by large businesses. Using centralities and node-link representations, they modeled these networks. They did not explicitly use data on actual population trajectories; instead, they used graphs that depict city roads/regions and traffic statistics [1]. Mahrsi et al. proposed a graph-based approach for clustering network-oriented trajectory data. However, this approach required preprocessing to map each point on the road which requires effort [2]. Hu et al. devised a mechanism that from road level segment, extracts urban functions and classifies it into necessary categories, and process them using graph convolutional neural network [3]. Sabarish et al. in their research used graph based hierarchical clustering method to fetch identical trajectories from a truck trajectory database [4]. Guo et al. considered analyzing trajectory network data to increase the accuracy of the location to avoid getting redundant data in trajectories of individuals and groups [5]. Huang et al. adapted the clustering of trajectories to analyze how people move in groups and the similarity between the trajectories [6]. Dai et al. devised a mechanism that used graph data structure to infer the shortest path within the road graph network based on the cost and also adapted an outlier detection method to filter the trajectory data [7]. Wang et al. studied how trajectory data is collected by transportation cells. They proposed a model which overcomes the problem of transportation cells catching the location when it passes through the cell and applying dynamic graph visualization to analyze the trajectory data [8]. Zhang et al. worked on the trajectories of a huge number of moving location points. Their research proposed a methodology to recognize patterns in such large trajectories using the clustering algorithm DBSCAN and then present it as a spatial graph [9].

III. Proposed Methodology

A. Dataset

The dataset selected for this research is the Microsoft Geolife dataset [15]. The Geolife dataset was collected for the city of Beijing over course of three years using the GPS location information of the users. The dataset contains over 17,000 trajectories and an approximate distance of over 1 million kilometers. The GPS location was recorded every 15 seconds which gives a spatio-temporal format of the trajectory data. Table I shows sample of dataset.

TABLE I
DATASET SAMPLE

Latitude	Longitude	Date_Time	Id_user	Label
39.895505	116.3162	06-04-2008 09:30	10	bus
39.895572	116.3212	06-04-2008 09:31	10	bus
39.853438	116.3576	07-12-2008 11:46	20	taxi
39.852103	116.3528	07-12-2008 11:47	20	taxi
39.98425	116.3201	18-09-2011 08:26	65	bike

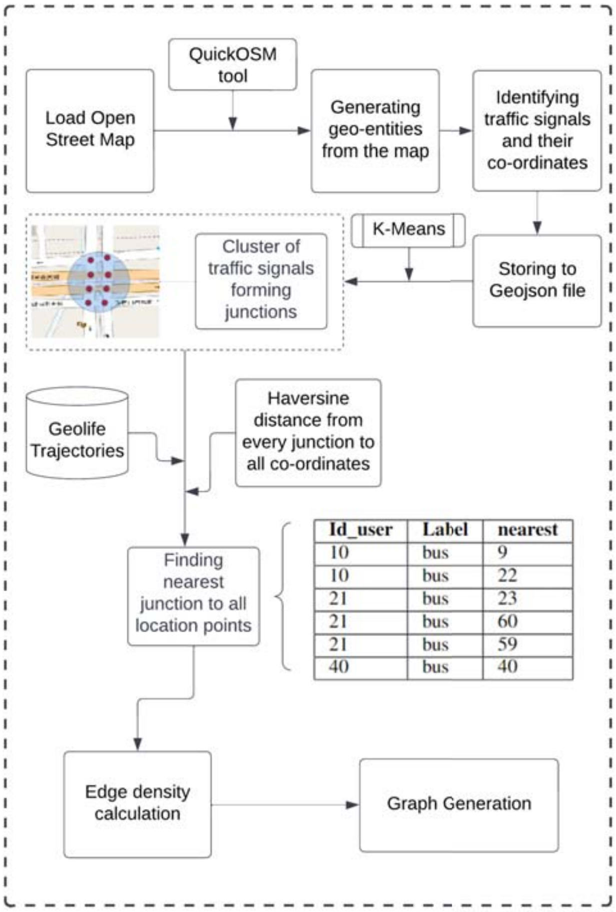


Fig. 3. Architecture of proposed system for graph generation

Fig. 4. Traffic signals geo-entities

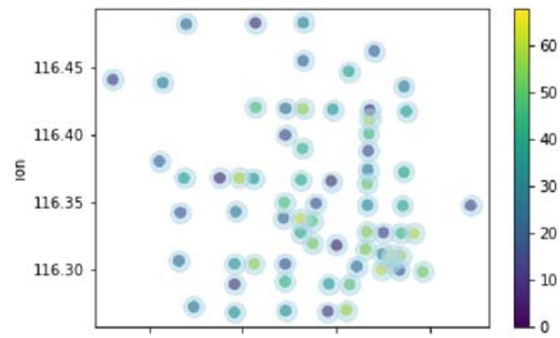


Fig. 5. Clusters of traffic signals.

B. Junction Identification

The first task was to convert the dataset into a junction-based trajectory dataset. We used OpenStreetMaps [18] to plot the dataset on the map. To extract the junction from the OpenStreetMap, we used the QGIS tool [12] to extract the locations of traffic signals from the map which is an inbuilt query in the QGIS tool. After extracting the traffic signal the problem arose as there were around 4-6 traffic signals per junction.

C. Clustering of signals

The QuickOSM plugin in the QGIS tool extracted the raw geo-entities of the traffic signals. As there were 4-6 signals per junction we used the K-means [11] clustering algorithm. We increased the number of clusters to a point where all the signals in one junction were under a single cluster. Then mean of all location coordinates in one cluster was taken which returned the center location co-ordinate of each junction. The junctions were then labeled accordingly. Fig 4 shows the OpenStreetMap before clustering. Fig. 5 shows the clusters of traffic signals formed as junctions.

D. Sequence of junctions

After junctions were identified, the dataset was plotted across the junctions. Haversine distance was used to identify the nearest junction. The Haversine distance formula [17] is crucial to navigation because it calculates the great-circle distances between two places on a sphere using their longitudes and latitudes. The Haversine distance (1) gave the nearest junction for each coordinate because a user in a vehicle at some point will have the closest location point to one of the junctions' coordinates, and the nearest coordinate of the user was recorded. This process was repeated for all trajectories of users and the data transition of users from junction to junction was recorded. Haversine distance is defined as:

$$D(x,y) = 2arcsin[\sqrt{sin^2((x_1-y_1)/2) + cos(x_1)cos(y_1)sin^2((x_2-y_2)/2)}] \quad (1)$$

The data obtained after after nearest junction calculation is shown in II

Remove redundant rows. [Fig. 7].

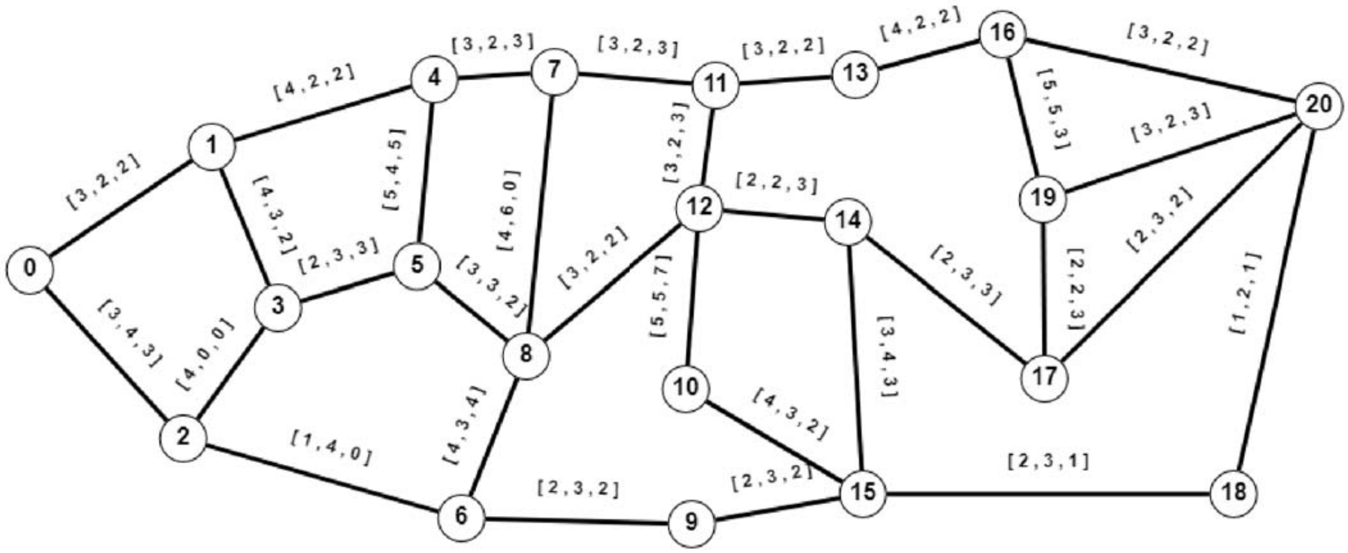


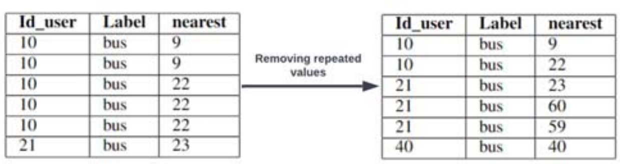
Fig. 6. Graph representation of the junctions identified with their edge density

TABLE II
NEAREST JUNCTIONS

Id_user	lat	lon	Nearest junction	Vehicle label
10	39.89551	116.3162	3	bus
10	39.89559	116.3172	3	bus
10	39.89559	116.3182	8	bus
21	39.8956	116.3196	8	taxi
22	39.89557	116.3212	6	bike

$$
\begin{array}{c|ccccc}
 & 0 & 1 & 2 & & n \\
\hline
0 & [4,2,2] & [0,0,0] & [8,2,0] & \cdots & [0,0,0] \\
1 & [0,0,0] & [0,0,0] & [6,0,1] & \cdots & [0,5,5] \\
 & \vdots & \vdots & \vdots & \ddots & \vdots \\
n & [0,0,0] & [4,5,0] & [0,0,0] & \cdots & [2,1,5]
\end{array}
$$

Edge Density Matrix



Id_user	Label	nearest
10	bus	9
10	bus	9
10	bus	22
10	bus	22
10	bus	22
21	bus	23

Id_user	Label	nearest
10	bus	9
10	bus	22
21	bus	23
21	bus	60
21	bus	59
40	bus	40

Fig. 7. Table after removing repeated values

Algorithm 1 Edge Density

---

m = 0, 1, and 2 are bus, $bike$, $taxi$ $respectively$

Let $M[n][n][m]$ be an edge density matrix of m vehicle labels for n vertices in a junction graph.

for i^{th} row in $Nearest\ junctions$ table **do**
  j_a , j_b = row[i][$nearest$] , row[i+1][$nearest$]
  m = row[i][$label$]
  $M[j_a][j_b][m]$ += 1
end for

E. Road edge density

After obtaining the road graph data we recorded the counts of different types of vehicles over the road graph network in a given time slot, which gave the vehicle density between two junctions which is referred to as edge in this paper. The calculated densities between two vertices are stored in a density adjacency matrix. As shown in the sample in Fig. 8, we can see the edge between vertices 1 and 4 has density $[6, 4, 5]$

After calculating the densities the proposed method for generating a junction-based road graph is completed. Now, this graph gives a lot of pieces of information like the vehicle density on the road segment, frequently traveled path, most often visited junctions, and much more.

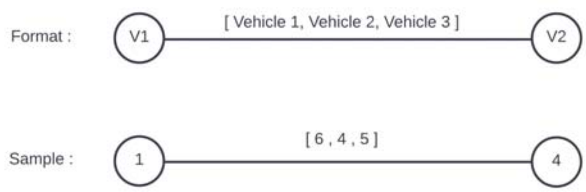


Fig. 8. Density representation on edge.

IV. Conclusion And Future Work

This study suggested a visual analytics method to analyze peoples' movement patterns in cities. The system preserves the original trajectory of users and employs a new junction-based graph model to depict the trajectory patterns and includes genuine traffic statistics, such as traffic density carried by each road segment and its vehicle type. The visualization of the road network as a graph along with the densities of vehicles can be used in urban planning to decide where to build new building structures, deploy ATMs, Emergency rooms, fuel stations, etc. This data can be used in several trajectory data mining, and since the data is a road network graph, several graph-based algorithms can be applied to extract information about the cities' road network and make development decisions based on that data.

References

[1] Rozenblat, Céline, and Guy Melançon, eds. Methods for multilevel analysis and visualisation of geographical networks. Springer Netherlands, 2013.
[2] Mahrsi, Mohamed Khalil El, and Fabrice Rossi. "Graph-based approaches to clustering network-constrained trajectory data." International Workshop on New Frontiers in Mining Complex Patterns. Springer, Berlin, Heidelberg, 2012.
[3] Hu, Sheng, et al. "Urban function classification at road segment level using taxi trajectory data: A graph convolutional neural network approach." Computers, Environment and Urban Systems 87 (2021): 101619.
[4] Sabarish, B. A., R. Karthi, and T. Gireesh Kumar. "Graph similarity-based hierarchical clustering of trajectory data." Procedia Computer Science 171 (2020): 32-41.
[5] Guo, Diansheng, Shufan Liu, and Hai Jin. "A graph-based approach to vehicle trajectory analysis." Journal of Location Based Services 4.3-4 (2010): 183-199.
[6] Huang, Yan, Cai Chen, and Pinliang Dong. "Modeling herds and their evolvements from trajectory data." International Conference on Geographic Information Science. Springer, Berlin, Heidelberg, 2008.
[7] Dai, Jian, et al. "Path cost distribution estimation using trajectory data." Proceedings of the VLDB Endowment 10.3 (2016): 85-96.
[8] Wang, Zuchao, et al. "Visual exploration of sparse traffic trajectory data." IEEE transactions on visualization and computer graphics 20.12 (2014): 1813-1822.
[9] Zhang, Junming, et al. "On retrieving moving objects gathering patterns from trajectory data via spatio-temporal graph." 2014 IEEE International Congress on Big Data. IEEE, 2014.
[10] Berke, Alex, et al. "Generating synthetic mobility data for a realistic population with RNNs to improve utility and privacy." Proceedings of the 37th ACM/SIGAPP Symposium on Applied Computing. 2022.
[11] Pham, Duc Truong, Stefan S. Dimov, and Chi D. Nguyen. "Selection of K in K-means clustering." Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science 219.1 (2005): 103-119.
[12] Graser, Anita. Learning QGIS 2.0. Packt Publishing Ltd, 2013.
[13] Zheng, Yu. "Trajectory data mining: an overview." ACM Transactions on Intelligent Systems and Technology (TIST) 6.3 (2015): 1-41.
[14] Acs, Gergely, and Claude Castelluccia. "A case study: Privacy preserving release of spatio-temporal density in paris." Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining. 2014.
[15] Yu Zheng, Xing Xie, Wei-Ying Ma, GeoLife: A Collaborative Social Networking Service among User, location and trajectory. Invited paper, in IEEE Data Engineering Bulletin. 33, 2, 2010, pp. 32-40.
[16] Gao, Zhigang, et al. "Protecting Location Privacy of Users Based on Trajectory Obfuscation in Mobile Crowdsensing." IEEE Transactions on Industrial Informatics (2022).
[17] Maria, E., E. Budiman, and M. Taruk. "Measure distance locating nearest public facilities using Haversine and Euclidean Methods." Journal of Physics: Conference Series. Vol. 1450. No. 1. IOP Publishing, 2020.
[18] OpenStreetMap contributors. OpenStreetMap database [PostgreSQL via API]. OpenStreetMap Foundation: Cambridge, UK; 2021 [cited 22 Dec 2021]. © OpenStreetMap contributors. Available under the Open Database Licence from: openstreetmap.org. Data mining by Overpass turbo. Available at overpass-turbo.eu.