Modeling and Querying Trajectories using Neo4j Spatial and TimeTree for Carpool Matching

Fuat Bakkal
Department of Computer Engineering
Kocaeli University
Kocaeli, Turkey
fuat41@gmail.com

Nurullah Samed Savaş
Department of Computer Engineering
Kocaeli University
Kocaeli, Turkey
nurullahsamed@gmail.com

Süleyman Eken
Department of Computer Engineering
Kocaeli University
Kocaeli, Turkey
suleyman.eken@kocaeli.edu.tr

Ahmet Sayar
Department of Computer Engineering
Kocaeli University
Kocaeli, Turkey
ahmet.sayar@kocaeli.edu.tr

***Abstract*—With the the exponential growth of location aware devices, analysis of human movements has been the subject of several studies. Problems related to urban mobility such as vehicle congestion are serious concern in cities. Carpooling is one of the solutions to soften congestion problem. This paper presents a novel matching method for carpooling. Trajectories are firstly modeled using Neo4j spatial and Neo4j TimeTree libraries. Then, temporal and locational filtering steps are operated. We extensively evaluate the efficiency and efficacy of the proposed system on Geolife trajectory dataset.**

***Keywords*—carpooling; moving objects; trajectory; location based services; Neo4j spatial; Neo4j TimeTree; human mobility behavior**

I. Introduction

Acquisition of a moving object’s positions has become technically possible thanks to the positioning sensors such as GPS (Global Positioning System) loggers and GPS-phones. Such kinds of devices enable the tracking of moving objects and generate a large amount of mobility data allocated in time and space and. Moving objects are considered as every physical object in the real world equipped with a device that allows the tracking of its geographical position at a given time. Typical examples of mobile objects are vehicles, people, animals and airplanes.

Mobile phones embedded with GPS, accelerometer, Internet connection, digital camera allow the creation of several mobile services called Location-Based Services (LBSs). The LBSs allow to use positioning information anywhere and anytime in order to provide new details about people, events and others [1]. Research communities have made tremendous research effort to support LBSs (e.g. mobile object databases). In addition to conventional search functions of moving objects, many LBS applications need to analyze and mine various moving patterns and phenomenon of tracked objects. Another type of service, Mobile Trajectory Based Services (MTBS), has become popular. MTBS is related to users’ mobility profiles, or simply users’ trajectories, which are fundamentally collections of mobile traces that can reveal moving patterns [2]. With the MTBSs, regular routes from a users’ historical trajectory dataset can be used for mining frequent trajectories [3], finding similar trajectories [4], mining points of interests (POIs), findout sub-trajectories, and also ridesharing recommendations to a group of users who share similar routes can be provided, and people's trip styles may be improved.

Urban mobility addresses the problems which are causing a lot of losses such as traffic congestion, environmental damages, health issues etc. Traffic congestion is one of the main problem that decreases the quality of urban mobility and it is a reality in most cities around the world. Many factors can cause to congestion in large cities, but the increasing number of vehicles can be identified as the main factor. The growth in the number of private cars has caused a lot of problems, indeed [5]. The traffic congestion has caused expressive economic losses and reflects on various aspect of society. Home delivery of foods, for instance, is directed affected by congestion and passes on the costs to the population [6]. Apart from economic losses, environment damages are a serious problem. An alternative to soften these problems and to improve the quality of urban mobility is to encourage the use of carpooling or ridesharing, which consists of sharing private vehicle space among people with similar destinations or daily trajectories.

In this paper, we firstly model the trajectory data of private vehicle owners (drivers) and hitchhikers for storage in Neo4j Spatial and TimeTree. Then, the best the results of a query that returns drivers are presented using carpooling matching strategy.

The remainder of this paper is organized as follows. Section II firstly presents literature review on carpooling.

978-1-5090-5795-5/17/$31.00 ©2017 IEEE

Section III explains the representation of trajectory data in Neo4j spatial and carpooling matching strategy. Experimental setup and performance results are given in Section IV. Section V concludes the paper and describes the directions for future work.

II. Related Works

Many studies have been proposed to address inefficiencies in the traffic system, and carpooling has been identified as an effective means of indirect traffic management, since if it is implemented effectively, it leads to the reduction of the number of vehicles participating in the transportation system. A detailed overview on the history of car pooling and how solutions have changed during the years, also based on available technology, can be found in [7].

There exists some software initiatives to facilitate the carpooling's practice; BlaBlaCar [8][9], lyft [10], Singu Carpooling [11], BeepCar [12] and Ridesharing [13] are some examples. Some services provided by these softwares require that interested users perform a search for people who offers a ride with the same or similar trajectories. Some of the essential features are the easy of use, safety, flexibility and efficiency. There are other important factors that may discourage carpooling: smoking, features of the vehicle itself, social and demographic profile of the driver and gender [14][15]. However, there is no easy way to filter trip colleagues according to their social or demographic profile. The so-called ridematching/carpool matching procedure has been proposed to deal with the these issues and suggest the carpooling formation instantaneously.

Most researchers have focused on trajectory mining process [16][17] for carpool routing. Furuhata et al. [18] integrate ridematching applications with social network a way to encourage people to use ridesharing application. Developing a carpool matching system requires the similarity between trajectories. There are some approaches in literature that calculate similarity such as LCSS (Longest common sub-sequence), EDR (Edit distance), Dynamic Time Warping (DTW). However, hese approaches consider all points of the trajectory which is very time consuming [19]. So, some researchers utilize trajectory discretization for reduction of computation of trajectory similarity. Cruz et al. [20]propose a system for clustering similar users trajectories in order to discover potential carpooling opportunities. Their approach includes Points of Interest (POI) as a method for trajectory discretization, temporal filtering and Optics algorithm for clustering. However, to our knowledge no previous study has been done on trajectory data in graph databases for carpooling purposes.

III. Methodology

In this paper, we focus on recommending hitchhiker to vehicles based on the drivers' weekly frequent routes as shown in Fig. 1. This recommendation is made by using the current and target position of the hitchhiker and time information. There are two different users in this system, a driver who owns a car and a hitchhiker asking for a ride. Sub-sections explain the main two parts of the system in detailed.

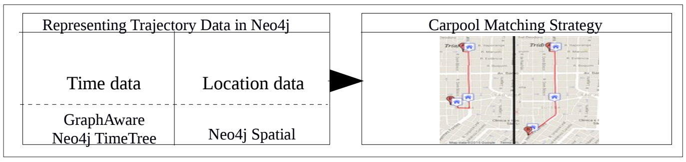

Fig. 1. Architecture of the proposed system

A. Representing Trajectory Data in Neo4j

Modelling and querying time and location based events like trajectories in a graph requires two different libraries in Neo4j: (i) GraphAware Neo4j TimeTree and (ii) Neo4j Spatial library. Our representation and query model is working in two stages. Firstly, building time tree which represents time varying temporal data and secondly, building spatial data tree which represents spatial data (latitude, longitude).

For time based data perspective, various approaches possible. We landed on a tree structure as the model that best fit the problem. The main key question we found ourself asking as we went through the process of building the time tree to connect the time series of locations in a trajectory. Finally we decided that it would be most effective to create the hour, minute and second level nodes. So, we used "GraphAware Neo4j TimeTree" library [21].

For spatial data, we used Neo4j Spatial library [22]. This library enables us to do spatial operations like spatial indexing and spatial querying (such as find things within a certain geometry). Since we are working with points (which refers to latitude, longitude set), we simply needed to create a "SimplePointLayer" to get access to point capabilities and proximity searches. To import our data correctly to the Neo4j, we initialized the spatial index, added a number of location points to the index. Fig. 2. shows the modeled trajectory data in Neo4j. Here, blue node indicates user, green node shows temporary node to link time and location nodes. Gray node stands for location data. Other nodes (pink, purple and red) indicate day, month, and year of one sampled trajectory data, respectively.

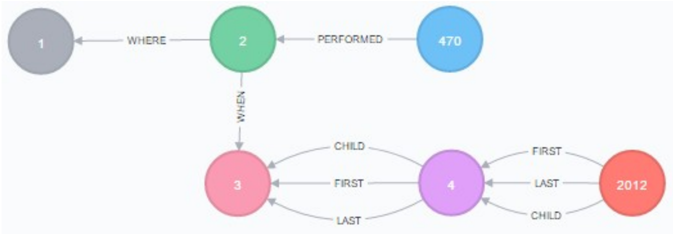

Fig. 2. Representation of trajectory data in Neo4j

B. Carpool Matching Strategy

After modeling trajectory data in Neo4j, it is needed to explain how carpool matching model works. Firstly, the hitchhiker has to specify the information of current and the target location when requesting for a vehicle. Then, several querying steps are run. Using current and the target location info, the first query is carried out for finding the available drivers. The first query is a filtering process on the trajectory information of the drivers. Within all drivers who have registered information on day of the week the hitchhiker requests the vehicle will be removed. The remaining data after the first step can be defined as GPS points, which consist of position and time information on the specified day of the week. Second query is operated using the current time and the current location on the remaining data. In this case, filtered GPS points are filtered at a time interval of one hour before and after the time the hitchhiker requests the vehicle and at a maximum distance of 100 meters from the user's current position. On this day, GPS points of vehicles are obtained that are close to the time of demand and have a distance of no more than 100 meters from current position of the hitchhiker on the day of the week which hitchhiker requests the vehicle. In the third query, for all vehicles, the time information of the nearest GPS points to the current position will be used. In the last filtering process, hitchhikers who have a time after the time information of the points close to the current position and who have a maximum distance of 100 meters to the target position will be taken. After three query steps, direction filtering is done for the reason why drivers have a time later than the time information of points near the current location of hitchhiker. With this condition, vehicles going from the target position to the current position is excluded from the query. As a result of these queries, on the day of the week when the hitchhiker requests the vehicle, vehicles will be obtained with GPS points at most 100 meters from both the current and the target position of the hitchhiker, and whose direction is from the current position to the target position. In addition, the number of days that these conditions are fulfilled will be determined. It is also determined which days are. By using this frequency information and the frequency of the drivers are presented to the hitchhiker for further communication between them. Next section gives experimental results of the proposed approach.

IV. Experimental Results

A. Datasets

In this paper, Geolife trajectory dataset which is created in (Microsoft Research Asia) Geolife project is used. It is collected by 182 users between April 2007 and August 2012 years. A GPS trajectory of this dataset is represented by a sequence of time-stamped points, each of which contains the information of latitude, longitude and altitude. This dataset contains 17,621 trajectories. These trajectories were recorded by different GPS loggers and GPS-phones, and have a variety of sampling rates. This dataset recoded a broad range of users' outdoor movements [23][24].

B. Experimentation setup

Experimental observations used the information of the first 20 users of the Geolife trajectory dataset and it is assumed that these belong to the drivers. In this data set, each user's data is stored separately on a daily basis. Within each day's data, GPS points consist of latitude, longitude, date and time information. For the experimental observation to be made on this information light, needed the current position of hitchhiker, the position information to be traveled and the time when the vehicle is requested. Once this information has been obtained from the user, available drivers are identified. Moreover, GPS points near the current and target position are displayed to the user/ hitchhiker taking into consideration the time information. The number of movements of drivers between these two points (current and target locations) will be presented to the user taking into consideration the direction. Number of movements, the date on which they were made also presented to the user taking into consideration the direction. Assume that current position of hitchhiker is 39.99984, 116.325001 has a request vehicle at 12:00 on 27/02/2017 and his/her target coordinates are 40.009328, 116.320887. In this case, firstly the day of the week is determined from the date information. The reason of this is vehicle users generally have a weekly routine. In this way, the processing load will be reduced by looking only specific data which given on the specified day of the week. After the reducing data size, the first query operation is performed using the user's current position and the time information of request. In this filtering, the GPS points that are within 100 meters from the current position of the user and within the time interval covering one hour before and after the time the user requests the vehicle are obtained. As a result, the remaining data will is as tabulated in Table I. The filtered dates specified here will only have GPS points that match the above distance and time condition.

TABLE I. Current location based filtered data

Filtered drivers id	Filtered Dates of Users
3	24/10/2008 – 15/11/2008 – 12/12/2008 - 23/02/2009
4	16/02/2009 - 06/03/2009- 01/05/2009 - 20/06/2009 - 27/06/2009
5	08/11/2008 – 23/02/2009

In the second filtering, GPS points having a distance of no more than 100 meters from the target position are obtained. It allows the condition to have a time later than the time information of the points near the current position in the now sufficiently miniaturized data set. The filtered data after the second query process is tabulated in Table II.

TABLE II. Current and target location based filtered data

Filtered drivers id	Filtered Dates of Users
3	15/11/2008
5	08/11/2008

In this way, vehicles passing from the current position of the hitchhiker at a time close to the time determined by the hitchhiker and passing from the target position in the direction from the current position to the target position are determined. In addition, the motion frequencies and dates of these vehicles are found. In Fig. 4, the GPS points after all these filtering operations are plotted on the map.

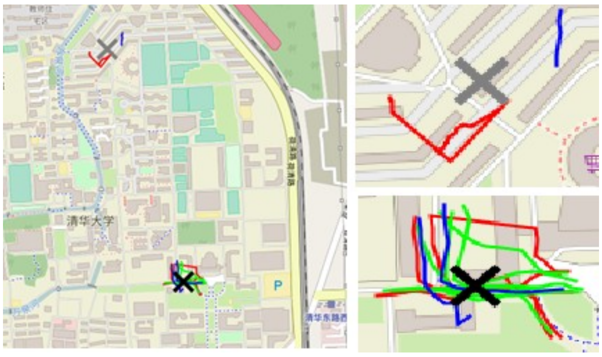

Fig. 3. Output of filtering operations

Where the black cross is the current position of the hitchhiker, and the gray cross is the target position. Trajectory of users with numbers 3, 4, 5 id are shown in red, green and blue, respectively. The same colored roads represent trajectories that are registered on different dates for a single user. Only user 3 and 5 are presented to the hitchhiker. Because driver 4 shown in green at the current location does not pass the target location of hitchhiker.

V. Conclusion

Car pooling can be described as the cooperation of two or more persons regarding the use of a single vehicle to meet their personal needs. In addition to the social benefits mentioned before, there are potential personal benefits for the individuals taking part into car pooling. These benefits range from reduced fuel costs, to reduced toll fees, reduced time spent on the road, etc.

Generally speaking, we provide the following contributions in this paper:

- We propose a carpooling system based on GraphAware Neo4j TimeTree and Neo4j Spatial library. The systems visualizes available drivers to hitchhikers via Google Maps and presents them in form of graph.
- We extensively evaluate the efficiency and efficacy of the proposed system on real-life trajectory dataset.

In the future, we will extend the proposed system with users profile (social and demographic information) in addition to trajectories. Also, graph based similarity of trajectories will be addressed.

Acknowledgements

We are thankful to Microsoft Research Asia for providing us Geolife dataset.

References

[1] E. Kaasinen, “User needs for location-aware mobile services,” Personal Ubiquitous Computing, 7, pp. 70–79, May 2003.

[2] S. Khokhar and A.A. Nilsson, “Introduction to mobile trajectory based services: A new direction in mobile location based services,” In Proc. of the 4th International Conference on Wireless Algorithms, Systems, and Applications, WASA ’09, pp. 398-407, Berlin, Heidelberg, 2009.

[3] N.S. Savage, S. Nishimura, N. E. Chavez, and X. Yan, “Frequent trajectory mining on GPS data,” In Proc. of the 3rd International Workshop on Location and theW eb - LocWeb ’10, pp. 1-4, 2010.

[4] N. Pelekis, I. Kopanakis, G. Marketos, I. Ntoutsi, G. Andrienko, and Y. Theodoridis, “Similarity Search in Trajectory Databases,” In Proc. Of 14th International Symposium on Temporal Representation and Reasoning (TIME’07), pp. 129–140, 2007.

[5] J.L. Funk, “It and sustainability: new strategies for reducing carbon emissions and resource usage in transportation,” Telecommunications Policy, 39, 10, pp. 861–874, 2015.

[6] D. Schrank, B. Eisele, T. Lomax, “Urban Mobility Report: Powered by INRIX Traffic Data, 2015.

[7] J. Xia, K.M. Curtin, W. Li, Y. Zhao, “A new model for a carpool matching service,” PLoS ONE, 10(6), 2015.

[8] Blablacar, http://www.blablacar.com, [Accessed: 06-March-2017].

[9] blablacar.co.uk, https://www.blablacar.co.uk/, [Accessed: 06-March-2017].

[10] L.G.J. Zimmer, Lyft, https://www.lyft.com/ [Accessed: 06-March-2017]

[11] Singu Carpooling, http://singucarpooling.com/, [Accessed: 06-March-2017].

[12] BeepCar, https://beepcar.ru/, [Accessed: 06-March-2017].

[13] Ridesharing, ttps://www.ridesharing.com/, [Accessed: 06-March-2017].

[14] V. Cezar Monteiro de Lira, Exploring Human Activity Behavior and Mobility Data in Carpooling, Master Thesis, Universidade Federal de Pernambuco, http://repositorio.ufpe.br/handle/123456789/11850, 2014.

[15] C. Michael Oliveira, On the similarity of users in carpooling recommendation computational systems, Master Thesis, Federal University of Sergipe, 2016.

[16] W. He, K. Hwang, D. Li, “Intelligent Carpool Routing for Urban Ridesharing by Mining GPS Trajectories,” 15(5): 2286–2296, 2014.

[17] J. Lee, J. Han, Trajectory Clustering: A Partition-and-Group Framework. 2007.

[18] M. Furuhata, et al., “Ridesharing: The state-of-the-art and future directions,” Transportation Research Part B: Methodological, 57, pp. 28–46, 2013.

[19] Y. Zheng, “Trajectory Data Mining: An Overview”, ACM Transactions on Intelligent Systems and Technology, 6(3): 1–41, 2015.

[20] M.O. Cruz, H. Macedo, A. Guimaraes, “Grouping similar trajectories for carpooling purposes,” in Proceedings of 2015 Brazilian Conference on Intelligent Systems, pp. 234-239, 2015.

[21] GraphAware TimeTree, http://graphaware.com/neo4j/2014/05/28/graph-aware-neo4j-framework.html [Accessed: 10-March-2017].

[22] Neo4j Spatial, https://github.com/neo4j-contrib/spatial [Accessed: 10-March-2017].

[23] Y. Zheng et al., “Mining interesting locations and travel sequences from GPS trajectories,” In Proceedings of International conference on World Wild Web, Madrid Spain. ACM Press: 791-800, 2009.

[24] Y. Zheng et al., “Understanding Mobility Based on GPS Data . In Proceedings of ACM conference on Ubiquitous Computing, Seoul, Korea. ACM Press: 312-321, 2008.