Proceedings of APCC2008 copyright (c) 2008 IEICE 08 SB 0083

Design and implementation of a movement history analysis framework for the taxi telematics system

Junghoon Lee
Dept. of Computer Science and Statistics
Cheju National University
690-756, Ara 1 Dong, Jeju-City, Republic of Korea
Email: jhlee@cheju.ac.kr

Gyung-Leen Park
Dept. of Computer Science and Statistics
Cheju National University
690-756, Ara 1 Dong, Jeju-City, Republic of Korea
Email: glpark@cheju.ac.kr

***Abstract*—This paper addresses a history data analysis framework for the Taxi telematics system, which keeps accumulating a great deal of movement data or location records from tens or hundreds of taxis in Jeju city. Each record consists of GPS-generated field, telematics-generated field, and framework-generated field. Aiming at boosting up diverse analysis on these data, the designed and implemented framework organizes the record into the database, defines a finite state machine to trace the status of each taxi, and provides a graphic interface, particularly for temporal movement change and trajectory visualization. We show that the framework enables us to efficiently extract such information as dispatch time, dispatch distance, traffic pattern, and so on. A new challenging service can be designed and tested using on our system.**

I. Introduction

[1] Telematics, the blending of computers and wireless telecommunication technologies, pursues the goal of efficiently conveying information over vast networks to improve a host of business functions or government-related public services[1]. Installed within a car, the telematics device can provide various LBSs (Location-Based Services) such as car navigation, vehicle tracking, and automatic collision notification[2]. The operating system for this in-vehicle device is gradually drawing attention of vendors. For example, *Windows Automotive* was released by Microsoft as another version of *Windows Mobile* operating system. This version can provide efficient running and developing environment for the LBS application[3].

Real-time location tracking, as the most fundamental service, makes a central server collect and manage information on the status and the location of a vehicle, driving behavior and preference of a driver[4]. This information can be visualized to a human manager or analyst, providing background data to diverse decision making. The vehicle types may be rent-a-car, taxi, truck, and so on. Particularly, for taxi company operating many taxis, real-time tracking provides the basic functional unit for an efficient taxi dispatch to a customer call, path recommendation based on the current traffic condition, saving the time and fuel[5].

[1]This research was supported by the MKE, Korea, under the ITRC (Information Technology Research Center) support program supervised by the IITA. (IITA-2008-C1090-0801-0040)

For location tracking, vehicle should be able to determine its location, typically represented by latitude and longitude, with GPS (Global Positioning System) receiver and then report to the central server via an appropriate wireless interface, be it infrastructure-based or ad-hoc style. The infrastructure-based cellular network such as 3G or CDMA (Code Division Multiple Access) in Korea, even though it needs non-negligible expense, can make each vehicle reliably report its location wherever it is moving on. Shorter report period, or frequent report may give more accurate and refined tracking data. However, due to the communication cost, the resolution of the tracking data is compromised with the cost negotiated with the telecommunication company.

The collected location data can also create many useful data on the distance and the time interval for passenger transaction as well as dispatch process, and other pick-up and drop-off behavior pattern[6]. In addition, this data can be used for developing and evaluating a new dispatch strategy, and also efficiently distributing company taxis according to the demand pattern analysis. In this regard, this paper builds a data analysis framework for such movement history information applicable both for on-line and off-line analysis, defines a finite state machine that traces the state change from taxis, and extracts information on taxi operation. It should be addressed that the research is built on top of the real data collected from the taxi telematics system currently in operation, and verifies the efficiency of the proposed analysis scheme with those data, showing the meaningful analysis result.

This paper is organized as follows: After issuing the problem in Section 1, Section 2 provides some background and related work. Then, Section 3 designs and implements a data analysis framework with a detailed description. After demonstrating the analysis results in Section 4, Section 5 summarizes and concludes this paper with a brief description on future work.

II. Background and related work

Jeju Island is a popular vacation spot for not just Koreans but also many international visitors, having many tourist attractions as well as a well maintained road network which essentially follows the entire coast (200 km) and crisscrosses[7]. By industrial and academic projects embarked from *Jeju Telematics City* enterprise, telematics devices are popularized for both rent-a-cars and taxis. Particularly, the in-vehicle telematics device in taxis contains a GPS receiver as well as an air interface, which follows CDMA protocol in Korea. At now, each vehicle reports its location and speed every minute with a reasonable communication cost negotiated with a telecommunication company. After all, Jeju area possesses a telematics network consist of a lot of active telematics devices, so in this area many challeging telematics services can be designed, developed, and tested.

Taxi telematics system has many sophisticated features to provide an advanced service to taxi drivers, customers and telematics researchers. Combined with global positioning system and radio communication technology, this system traces the position of taxis, finds a time saving route between start and destination points, dispatches the nearest taxi to the service call point based on the latest traffic information, and finally decides an efficient route for multiple destinations. It also developed a parallel computing framework based on a Linux cluster to host emerging telematics services that need intensive computing. For the efficient path recommendation, various versions of path finding schemes have been developed including A*[8], 1-to-1 and 1-to-may Dijkstra algorithm.

As for the analysis of movement history, *Project Lachesis* has proposed a number of rigorously defined data structures and algorithms for analyzing and generating location histories[9]. In their approach, *stays* are instances where a subject has spent some time at a single location, while *destinations* are clusters of stays. Based on this classification, the investigated and applied to two probabilistic models, one with and one without first-order Markovian conditioning. Even though this approach seems to be applied to a variety of analysis methods, it did not consider the specialty of taxi trajectory, where the stay or destination may be meaningless. J. Krumm has also proposed a new technique for enhancing a sequence of geographic points by tagging with nearby POI (Points Of Interest), associating them to Web pages. They used this tracking to estimate the link speed based on the Kalman filter method[10].

III. Proposed analysis framework

Figure 1 outlines the overall procedure for location data processing. Currently, up to 200 taxis report their location records every minute, and each record includes taxi ID and status fields in addition to the basic GPS data such as timestamp, latitude, longitude, direction, speed.

Map match module searches the link closest to the received coordinate from the digital map, and calculates the position ratio in the link and stores in *pos* field. This field indicates how close is the coordinate to the intersection, adding more meaningful information. For example, if this field gets closer to 0.0 or 1.0, namely, the point is located close to an intersection, the certainty of map matching result diminishes. In addition, if a vehicle is not moving at such spot, it must be waiting for the traffic signal change. After all, the location record is stored in the database according to the table structure shown in Table 1.

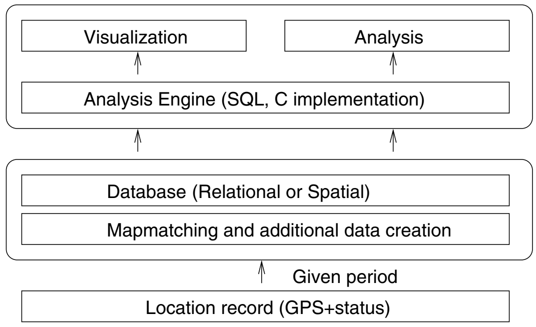

Fig. 1. Record processing architecture

TABLE I
Location record organization

field	type	description
tstamp	datetime	time
tid	char(6)	taxi ID
x	double(11,10)	latitude
y	double(11,10)	longitude
dir	double(3,2)	direction
speed	double(3,2)	speed of vehicle
status	int(2)	current taxi status
link	int(6)	matched link
pos	double(3,2)	position ratio

The most important field for the analyzing the record is the status of each taxi. It can have four values, namely, *empty*, *passenger-on*, *dispatched*, and *rest*. The record is generated at the telematics device interconnected to a taximeter and some manual switch. *passenger-on* and *empty* status is automatically set by this device. However, Returning from this status to *empty* should be manually set by the driver. This case happens when a taxi is dispatched to a customer call, but failed in picking up the caller. Sometimes, the driver possibly forgets this responsibility, so we should filter out such a imperfect data.

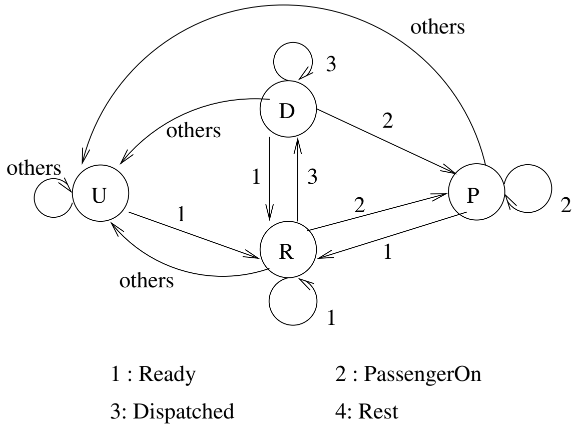

Fig. 2. State diagram of each taxi

To the end of efficient tracing the status of each taxi, we define a FSM (Finite State Machine) consist of 4 states, namely, *Undefined, Dispatched, Ready, and Pick-up*, respectively, for each taxi, and each location report changes the state. When a device is reset or stays too long in one state, especially in *Dispatched* or *Pick-up*, the state goes to *Undefined*, where the FSM waits for *empty* report to arrive. The FSM stays in the *Ready* state until it receives *dispatched* or *passenger-on* event. In the *Dispatched* state, a taxi may fail in picking up a customer. Then, the state returns to *Ready* by the arrival of *empty* message. If the taxi finally picks up the caller, the state transits to *Pick-up* state. Here, in ready state, if a taxi does not report any message for the last 1 hour, the FSM decides that the taxi has finished its operation session. In addition, if FSM stays in the *Dispatched* state for more than 20 minutes, the state is forced to change to the *Undefined* state.

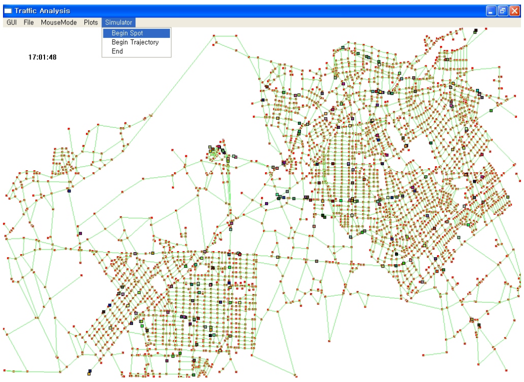

Fig. 3. Location tracking

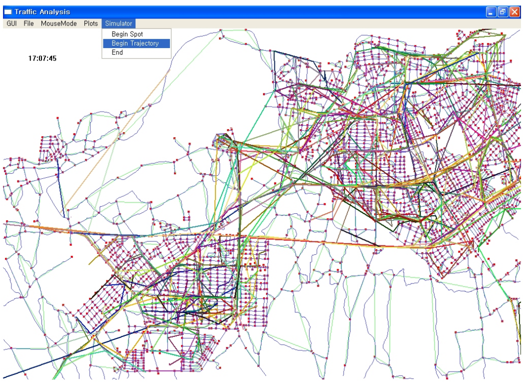

Fig. 4. Trajectory tracking

Figure 3 exhibits the visualization result of the location tracking of our framework. It shows the location of vehicles as a small rectangle on the city map. This figure is a snapshot of overall time series analysis. In addition, Figure 4 displays the graphic interface for the trajectory tracing. The movement of each vehicle is plotted as a straight line to give a visual understanding on the vehicle's trajectory. Finally, any retrieved points can be marked on the map, such as pick-up points, drop-off locations, and frequent call points.

IV. Analysis Results

To justify the effectiveness of our framework, this section performs an analysis on dispatch time and distance[11]. First, we measure the time interval during which a taxi stays in the *Dispatched* state. As the period of taxi report is 1 minute, this measurement inevitably includes an estimation error by up to 1 minute. The dispatch time is a very important performance criteria, as it indicates the time amount that a customer waits since he called a taxi. Figure 5 plots the probability density distribution for the dispatch time. As shown in this figure, most dispatch is accomplished within 13 minutes, while the average value being 7.670 minutes.

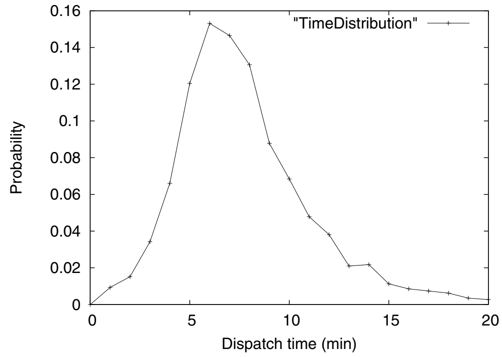

Fig. 5. Dispatch time distribution

In addition, the proposed framework also enables us to evaluate the taxi dispatch algorithm, as it can estimate the distance a taxi moves during the dispatch state. In this case, we cannot avoid the estimation error, as the exact path taken by the taxi cannot be known to the analyzer. Figure 6 illustrates this situation. A taxi has actually taken a route marked with a dotted line which follows the road network. However, as the report arrives discretely, the path can be inferred just by the sequence of non-continuous points. Hence, there is an estimation error between the length of dotted and solid lines. Even if we take the shortest path between the two points, it is also just another estimation. The vehicle may have taken another path. After all, our analysis work calculates the Euclidean distance between the points in distance estimation.

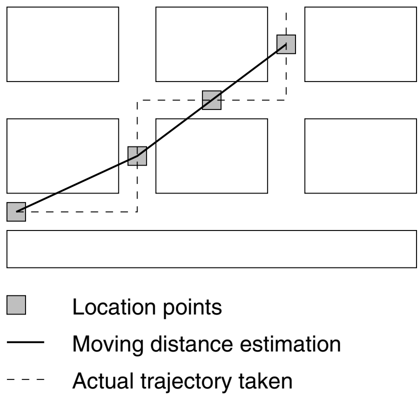

Fig. 6. Error in distance estimation

As plotted in Figure 7, the most distance distribution lies in less than 4 *kms*, showing the reasonable taxi dispatch performance in terms of distance criteria. However, it should be mentioned that the short distance is not the only performance requirement. Sometimes, the fairness among the taxis should be considered along with the dispatch time and distance. Our framework can easily measure such a constraint, so provides a useful environment for developing and testing a new dispatch algorithm.

V. Conclusion

This paper has presented a movement history analysis framework for *Taxi telematics* system currently running

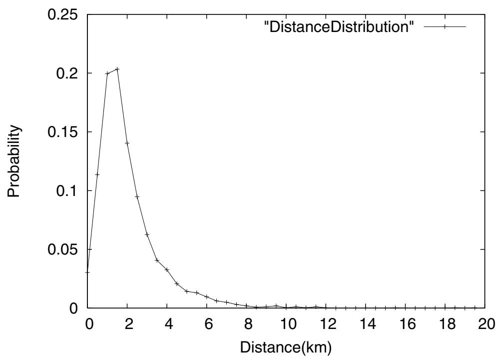

Fig. 7. Dispatch distance distribution

in Jeju city. The system is accumulating a great deal of records consist of GPS-generated field, telematics device generated field, and framework-generated field. The designed and implemented framework has organized the record into the database, defined a finite state machine to trace the status of each taxi, and provided a graphic interface. We justified the effectiveness of our framework by extracting such information as dispatch time and dispatch distance. After all, a new challenging service can be designed and test on our system, inviting a new challenging service including future traffic estimation[12], trajectory clustering[13], and so on.

References

[1] http://en.wikipedia.org/wiki/Telematics

[2] J. Lee, E. Kang, G. Park, "Design and implementation of a tour planning system for telematics users," *Lecture Notes in Computer Science*, Springer Verlag, Vol. 4707, 2007, pp.179–189.

[3] http://www.microsoft.com/windowsautomotive/default.mspx

[4] M. Imaura, K. Kobayashi, and K. Watanabe, "Real-time positioning by fusing differential-GPS and local vehicle sensors", *SICE Annual Conference*, 2003.

[5] S. Kiruthivasan, C. Deepakumar, and S. Althaf, "Decision Support System For Call Taxi Navigation Using GIS-GPS Integration", *MAP India*, 2006.

[6] P. Green, "Driver Distraction, Telematics Design, and Workload Managers-Safety Issues and Solutions," *Proceedings of the 2004 International Congress on Transportation Electronics*, 2004, pp.165–180.

[7] J. Lee, G. Park, H. Kim, Y. Yang, P. Kim, S. Kim, "A telematics service system based on the Linux cluster," *Lecture Notes in Computer Science*, Springer Verlag, Vol. 4490, 2007, pp. 660–667.

[8] A. Goldberg, H. Kaplan, and R. Werneck, "Reach for A*: Efficient point-to-point shortest path algorithms", *Microsoft MSR-TR-2005-132*, 2005.

[9] R. Hariharan and K. Toyama, "Project Lachesis: Parsing and modeling location histories", *Lecture Notes in Computer Science*, Springer Verlag, Vol. 3234, 2004, pp.106–124.

[10] J. Krumm, "The geographic context browser", *International Workshop on Exploiting Context Histories in Smart Environments*, 2005.

[11] Z. Liao, "Real-time taxi dispatching using global positioning systems", *Communication of the ACM*, 2003, pp.81–83.

[12] S. Lee, B. Lee, and Y. Yang, "Estimation of Link Speed Using Pattern Classification of GPS Probe Car Data", *Proc. International Conference on Computational Science and its Applications*, 2006, pp.495–504.

[13] S. Kim, J. Won, J. Kim, M. Shin, J. Lee, and H. Kim", "Path prediction of moving objects on road networks through analyzing past trajectories", *Lecture Notes in Computer Science*, Vol. 4693, 2007, pp. 379–389