To Taxi or Not To Taxi? - Enabling Personalised and Real-Time Transportation Decisions for Mobile Users

Wei Wu, Wee Siong Ng, Shonali Krishnaswamy
Data Mining Department
Institute for Infocomm Research, A*STAR, Singapore
Email: wwu, wsng, spkrishna@i2r.a-star.edu.sg

Abhijat Sinha
Faculty of Information Technology
Monash University, Australia
Email: callmerahul20@gmail.com

Abstract—We demonstrate a system that monitors the taxi availability at taxi stands by mining real-time taxi trajectory data streams. The system includes a server-side trajectory data stream processing and mining program and a client-side mobile application for Android smartphones. The server program continuously monitors for each taxi stand the numbers of taxis queuing at the taxi stand, the numbers of taxis that will pass the taxi stand, as well as the traffic conditions in the area around the stand. It delivers real time taxi and traffic information to mobile users via RESTful web services. The client-side location-based mobile application consumes these services to help mobile users make informed transportation choices. For example the availability of taxis might yet be a deterrent when traffic is congested. Real world taxi trajectory data from more than 14000 taxis are used in the demo.

I. INTRODUCTION

People in big cities rely on public transportation (such as buses, trains and trams) and taxis in their daily work and life. Very often commuters need to make choices regarding their transportation mode based on constraints such as time to reach destination and so on. There are very few mechanisms for commuters to obtain the necessary information in a ‘one-stop-shop’. Thus, commuters need to know in real tim the following:

- Are there taxis available at nearby locations?
- Are there taxis passing by?
- Is the traffic busy?

Another problem people often face is where they should take a taxi from. This typically happens when a person is already on bus or subway. For example, a commuter on subway may suddenly realize that she has an important meeting to attend. She decides to get off the subway and take a taxi to catch the meeting. The question is where she should get off to find a taxi. In such a case, she would like to alight at a subway station where there is available taxi and the traffic is smooth.

To enable mobile users make informed transportation decisions, we have developed a system that delivers real-time taxi availability and traffic information to users’ mobile phones. Our system continuously monitors the number of taxis queuing at taxi stands and the number of available taxis that will pass the taxi stands soon. Furthermore, it also monitors the traffic condition in the areas around the taxi stands. With real-time information of taxi availability and traffic conditions, people can decide whether they should take bus/subway first and where is the best place for them to get a taxi.

Such a system is feasible nowadays because most taxis are now equipped with GPS devices and most users have mobile phones. For example, Singapore has more than 14000 taxis carrying GPS devices and the taxis send location and status to a central server every certain seconds, and the mobile phone penetration in Singapore is 145 percent (many people carry more than one mobile phone) and 72 percent of mobile phones in Singapore are smartphones [1]. Our system is built based on existing infrastructure.

Our system gets real-time taxi availability information and traffic condition data by processing and mining real-time taxi trajectory data streams. Spatial range queries [2] are used to facilitate the moving object data stream mining tasks. To make our system be able to handle large number of taxis and taxi stands, in-memory spatial indexing [3] is employed to speedup the processing of trajectory data streams.

Our system also includes a location-based mobile application (for Android smartphones) that presents taxi availability and traffic information to mobile users. It queries our server based on user’s current location and visualizes taxi stand locations, number of available taxis, traffic condition on a map.

Compared to many existing taxi information systems (such as Taxi Magic [4] and TaxiBeat [5]), our system is different in terms of both technology and information delivered to users. Our system integrates trajectory data stream mining, continuous spatial query processing, web services, and mobile application. Existing taxi information systems only help a user book a taxi. Our system helps users make decision on whether it is a good idea to take taxi at the user’s current location.

In the demo of our system, we use real-world taxi trajectory data collected from more than 14000 taxis in Singapore. We show that our system can process the trajectory data stream to extract taxi availability and traffic speed information. We also demo that our mobile application can retrieve this information based on user’s current location and show it on a map to help users make informed decisions.
II. RELATED WORK

There has been much work on spatial queries on trajectory data (for example [6]) and on trajectory mining (for example [7]). Here we only briefly survey some work on taxi trajectory analysis and mining.

Marco Veloso et al. in [8] present a spatiotemporal analysis of taxi GPS traces. They identify popular pick-up and drop-off places, and examine the temporal variation of service. They also study the distribution of trip distance as well as the relationship between driving strategy and driver’s income.

Li Xiaolong et al. [9] use taxi trajectory data to find out hotspots where there are a lot of taxi demands. They also study how to predict future taxi demands at these hotspots.

In [10] Yu Zheng et al. demonstrate a system called T-Drive that learns time dependent fastest path to a destination from taxi drivers’ historical trajectories. In [11] information from other sources is further incorporated when the system suggests driving routes to users.

Bastani et al. [12] propose a system called flexi that mines combinable trips from taxi trajectories. The purpose is to suggest mini-shuttle like transportation system that is faster than bus or subway but more environment friendly than taxis.

Jing Yuan et al. present in [13] a recommendation system for both taxi passengers and taxi drivers. The proposed system suggests users some nearby locations where they are more likely to get taxis. It also suggests taxi drivers some locations where they are more likely to get passengers.

Our work differs from existing work in the following:

- We focus on extracting real-time taxi availability and traffic speed information from taxi trajectory streams.
- Our system is designed for user decision making on transportation. It is beyond where to find a taxi. Rather, it is more about whether to take a taxi or not.
- Our system can potentially enable sustainable transportation behaviors in the longer term.

III. TAXI AVAILABILITY AND TRAFFIC CONDITION MONITORING

Figure-1 is the architectural overview of our system. Taxis send their trajectory data stream to a server via wireless communication. Our program running on the server processes the taxi data streams and continuously updates the taxi availability at the taxi stands. It also computes traffic speed based on taxi location and speed data.

The server provides RESTful web services for mobile clients to retrieve the taxi and traffic information.

Mobile application running on a user’s mobile phone connects to the server and retrieves the relevant information based on user’s location location.

In this section we describe the taxi trajectory stream data and our server-side program. We will present the mobile client application in Section IV.

A. Real-Time Taxi Trajectory Data Stream

Each taxi participating in the system is equipped with GPS unit. The taxi has a device that sends (via wireless communication) the following information of the taxi to the server every certain seconds: \(< id, t, loc, v, s >\) where

- \(id\) is the unique identifier of the taxi,
- \(t\) is the time,
- \(loc\) is the taxi’s current location including longitude and latitude,
- \(v\) is taxi’s current speed, and
- \(s\) is taxi’s current status (such as FREE, OCCUPIED and so on).

B. Taxi Stands

We model a taxi stand as a location point and three regions \(< p, rgn_1, rgn_2, rgn_3 >\). The point \(p\) is the taxi stand’s location and the regions are geographical areas that we will monitor for the taxi stand. Specifically, \(rgn_1\) is the taxi stand’s taxi queuing area. \(rgn_2\) is the area covering the roads connecting to the taxi stand. If a taxi is running in this area it will pass the taxi stand. \(rgn_3\) is the area where we monitor the traffic speed around the taxi stand. Typically \(rgn_1\) contains \(p\), \(rgn_2\) is close to \(rgn_1\), and \(rgn_3\) contains \(rgn_2\) and \(rgn_1\).
Figure 2 shows an example of a taxi stand and the monitoring regions. The location point is shown as a balloon. The taxi stand region \((\text{rgn}_1)\) is shown as a blue polygon containing the taxi stand location, and the red polygon \((\text{rgn}_2)\) is the region where we monitor the number of taxis that will pass the taxi stand. The traffic monitoring region \((\text{rgn}_3)\) is not shown in the figure because it is bigger than the figure boundary.

Because of two reasons, we need to use \(\text{rgn}_2\) to monitor the number of taxis that will pass the taxi stand. Firstly, only monitoring the number of queueing taxi (in region \(\text{rgn}_1\)) may not be enough, because many taxis do not go to queue in the taxi stand, especially when there is no passenger waiting at the taxi stand. Secondly, the GPS sampling rate on the taxis may be low (for example a taxi may send a location update message every one minute), therefore the taxis queueing not long enough in \(\text{rgn}_1\) may not be captured.

C. Monitoring Engine

The job of the monitoring engine is to process the taxi trajectory data streams to find out taxi availability and traffic speed for the taxi stands.

We leverage continuous spatial queries to process the data. For each taxi stand, we introduce three continuous spatial queries with the three monitoring regions associated with the taxi stand. The queries collect the stream tuples in their spatial regions.

The purpose of the query with monitoring region \(\text{rgn}_1\) is to collect stream tuples for us to count the number of available taxis queueing in the taxi stand. For this reason, the query runs on the latest stream tuples and only tuples from taxis whose status is FREE are counted.

For the query with monitoring \(\text{rgn}_2\), we associate it with a moving time window, for example three minutes, so that it collects stream tuples in the last three minutes. We need to introduce a time window for this spatial query because we are interested in the number of taxis that passed the taxi stand in the last few minutes. Without a time window, the query result will only contain taxis currently in that region, and most of the time the result may be empty. It is a lot more informative to tell the user the number of the taxis passing the taxi stand in the last few minutes so that they roughly know the expected waiting time. In other words, we use the number of taxis passing the taxi stand in the last few minutes to estimate the number of taxis that will pass the taxi stand in the next few minutes. Only tuples from taxis whose status is FREE are counted.

For the query with monitoring \(\text{rgn}_3\), we also associate it with a moving time window. The purpose is to use the result records in the time window to compute the average traffic speed in the region in the last few minutes. In this case, taxis are used as traffic speed sensors. To make the calculated traffic speed accurate and representative, we filter out the taxis that are not in normal driving mode, such as taxis in PAYMENT status (the passenger has arrived her destination and is making payment). Such taxis do not represent the normal traffic flow in the area.

D. Web Services

Our server provides RESTful web services for mobile application to retrieve the taxi stand locations, taxi availability information at the taxi stands, as well as the traffic condition around the taxi stands.

IV. MOBILE APPLICATION

To help mobile users easily retrieve the taxi availability and traffic condition information, we build a location-based mobile application for them to use on their smartphones. The application consumes the RESTful web services exposed by our server. Figure 3 shows some screenshots of our mobile application running on an Android mobile phone.

The mobile application is able to get the list of taxi stands close to user’s current location. The user can also input an address to retrieve taxi stands close to that address, as shown
Fig. 3. Screenshots of the mobile application in our system.

(a) Search interface  
(b) Searching for taxi stands based on user’s location  
(c) Show taxi stands and traffic condition on map  
(d) Detail information at a taxi stand

in Figure 3(a). The retrieved taxi stands are overlayed on a map. The application also retrieves the traffic speed around the taxi stands. Colored circles are used to indicate the traffic speed. Figure 3(c) shows an example. Here red circle indicates that the traffic there is slow, and yellow means fair and green means fast. The user’s current location is also shown on the map (as a red balloon).

When a user clicks on a taxi stand, the application retrieves the detailed taxi availability and traffic information from the server and shows it to the user. Figure 3(d) shows an example. Number of taxis queueing at the taxi stand, number of available taxis that will pass the taxi stand in the next few minutes, and the traffic speed around the taxi stand are shown. Based on this information, a user can decide whether she wants to take a taxi at that taxi stand. The application is also able to find out (from Google Map service) walking direction from user’s current location to the selected taxi stand and show it to the user. Figure 3(d) shows an example.

V. DEMONSTRATION

A. Demo Setup

One PC (or laptop) and one Android smartphone are used in the demo. The PC serves as a server and the mobile phone works as a mobile client. Wireless communication is required for the mobile phone to connect to the PC to access the web services.

B. Taxi Trajectory Dataset and Taxi Stand Dataset

We use real-world taxi trajectory data in the demo. The trajectory data were collected from more than 14000 taxis. Every second the server program receives and processes hundreds to thousands of stream tuples.

We have collected real-world taxi stand information close to the subway stations in our city. We also associated monitoring regions with each of them.

C. Demo scenario

During the demo, a user traveling on the subway is simulated. She uses her smartphone to find out the location of taxi stands along the subway line. She checks the taxi availability at the taxi stands and the traffic condition in these areas. Based on this real-time information, she decides where she should alight to take a taxi.

ACKNOWLEDGMENT

This work was supported by A*STAR grant number 1021580037.

REFERENCES