IOSR Journal of Computer Engineering (IOSR-JCE)
ISSN: 2278-0661, ISBN: 2278-8727, PP: 20-24
www.iosrjournals.org

GPS Trajectories Based System: T-Finder

Ms. Chavan Mayuri[1], Mr. Gade Rajesh[2]
[1]*(Assistant Professor, Dept. of Information Technology, Dr. J.J. Magdum College of Engg., Jaysingpur, India)*
[2]*(Assistant Professor, Dept. of Information Technology, KIT's College of Engg., kolhapur, India)*

ABSTRACT: *This paper presents a recommender system for both taxi drivers and people expecting to take a taxi, using the knowledge of 1) passengers' mobility patterns and 2) taxi drivers' picking-up/dropping-off behaviors learned from the GPS trajectories of taxicabs. The objective of this paper is, 1) it provides taxi drivers with some locations and the routes to these locations, towards which they are more likely to pick up passengers quickly (during the routes or in these locations) and maximize the profit of the next trip. 2) It recommends people with some locations (within a walking distance) where they can easily find vacant taxis. In our method, we learn the above-mentioned knowledge (represented by probabilities) from GPS trajectories of taxis. We feed the knowledge into a probabilistic model which estimates the profit of the candidate locations for a particular driver based on where and when the driver requests the recommendation.*
Keywords- *Location-based services, recommender systems, trajectories, taxicabs, parking place detection*

1. Introduction

Many time in metro cities we usually suffered from waiting a long time for a taxicab. Actually, taxi drivers are also upset when cruising on road surfaces for finding passengers. An urgent challenge of current system is the vacant taxis cruising on roads do not only waste gas and time of a taxi driver but also generate additional traffic in a city. Thus, how to improve the utilization of these taxis and reduce the energy consumption effectively poses. Recently, in many big cities, like New York, Beijing, and Singapore, taxicabs are equipped with GPS sensors for dispatching and safety. Typically, these taxis will report on their present locations to a data center in a certain frequency, e.g., 2 minutes [6]. Besides a geo-position and time stamp, the occupancy information of a taxi is also recorded (using some weight sensor or by connecting a taxi meter with the embedded GPS device). Therefore, a large number of GPS trajectories with occupancy information are being generated every day. Intuitively, these taxi trajectories contain two aspects of knowledge. One is passengers' mobility, i.e., where and when passengers get on and off a taxi. The other is taxis' pick-up/drop-off behaviors. For example, where high-profit taxi drivers usually go and how they can find passengers quickly.

With these two aspects of knowledge, the author has presented a recommender system for both taxi drivers and passengers using a huge number of historical GPS trajectories of taxis. Specifically, on the one hand, given the geo-position and time of a taxicab looking for passengers, we suggest the taxi driver with a location, towards which he/she is most likely to pick up a passenger as soon as possible and maximize the profit of the next trip. This recommendation helps reduce the cruising (without a fare) time of a taxi thus saves energy consumption and eases the exhaust pollution as well as helps the drivers to make more profit.

In this paper, the authors have provided the system with people expecting to take a taxi with the locations (within a walking distance) where they are most likely to find a vacant taxicab. Using our recommender system, a taxi will find passengers more quickly and people will take a taxi more easily thereby reducing the supply/demand disequilibrium problem to some extent.

2. System Description and Design

In the proposed system, it improves the utilization of these taxis and reduces the energy consumption effectively poses an urgent challenge. The basic proposed system includes an approach to detect parking places based on a large number of GPS trajectories generated by taxis, where the parking places stand for the locations where taxi drivers usually wait for passengers with their taxis parked. For that purpose some calculations are made regarding a probabilistic model to formulate the time-dependent taxi behaviors and enable a city-wide recommendation system for both taxi drivers and passengers. But after different discussions and calculations to improve the taxi recommender by considering the time-varying queue length at the parking places; we approach to enhance the passenger recommender by estimating the waiting time on a specified nearby road segment in addition to calculating the probability of finding a vacant taxi.

The main advantage of this design is that, we will have a taxi-passenger recommender system based on the pick-up behaviors of high-profit taxi drivers and the mobility patterns of passengers learned from a large number of taxi trajectories.

3. Preliminary and Assembly Work

3.1 Road Segment: A road segment r is a directed edge that is associated with a direction symbol r.dir (one-way or bidirectional), two terminal points r.s and r.e, road level r.level (e.g., level-0 roads are mainly highways), as well as the travel time r.t.

3.2 Route: A route R is a sequence of connected road segments, i.e., R: $r_1 \rightarrow r_2 \rightarrow \cdot\cdot\cdot \rightarrow r_n$, where $r_{k+1}.s = r_k.e$, $(1 \leq k < n)$. The start point and end point of a route can be represented as R.s = r1.s and R.e = r_n.e.

3.3 State: We consider three states for a working taxi: occupied (O), cruising (C) and parked (P), detailed in Table 1. The taxi is non-occupied for both the cruising and parked states.

3.4 Trajectory and Trip: A taxi *trajectory* is a sequence of GPS points logged for a working taxi, here each point *p* has the following fields: time stamp *p.t*, latitude *p.lat*, longitude *p.lon*, located road segment (provided by map matching [7]) *p.r*, state *p.s* (The raw GPS trajectory only indicates whether a point is occupied or non-occupied). A taxi *trip* is a sub-trajectory which has a single state, either cruising (need to be inferred) or occupied. Note that a taxi could generate multiple trips between two parking places.

Since the concept gives brief idea about utilizing waste heat at domestic level, hence we have decided to use a “Voltas” second hand working domestic refrigerator of capacity 165 liters. Parts of domestic refrigerator are as follows.

Compressor, Modified Air cooled Condenser, Capillary Tube, Plate type Evaporator & Insulated Cabin.
The insulated cabin is a peripheral component which is used for utilizing the waste heat from refrigerator. This insulated cabin is fabricated by using galvanized iron sheets.

Table 1: The states of a taxi

Sr. No	**states**	Taxi Status
1.	Occupied (*O*)	A taxi is occupied by a passenger.
2.	Cruising (*C*)	A taxi is traveling without a passenger.
3.	Parked (*P*)	A taxi is waiting for a passenger.

Note that the “parked” state proposed in this paper is the status that taxi drivers wait somewhere for business, i.e., stay and/or queue for a while with the intention to get a passenger on-board. This status is frequently found at airports, hotels, shopping centers, etc. We call the places where the taxis are often parked as *parking places*. The parking place here does not merely imply a parking lot for private vehicles (which is the typical definition for the “parking place”).

4. Framework

The framework of the system is illustrated in Figure 1.An approach is to detect the parking places from GPS trajectories and segment the GPS trajectories according to Definition 4, then map-match the GPS trajectories to road networks using the IVMM algorithm [7], which outperforms other approaches for low-sampling-rate GPS trajectories. Later, we will utilize the detected parking places and the mapped trajectories to learn the time-dependent statistical results based on a probabilistic model. To tackle the data sparseness problem, we devise a road segment clustering method and perform statistical learning on each road segment cluster instead of a single road segment. The above processes will perform offline and will be repeated only when the trajectory data is updated (e.g., once a month). Based on this model, we perform recommendations to taxi drivers and passengers, given their locations and current times.

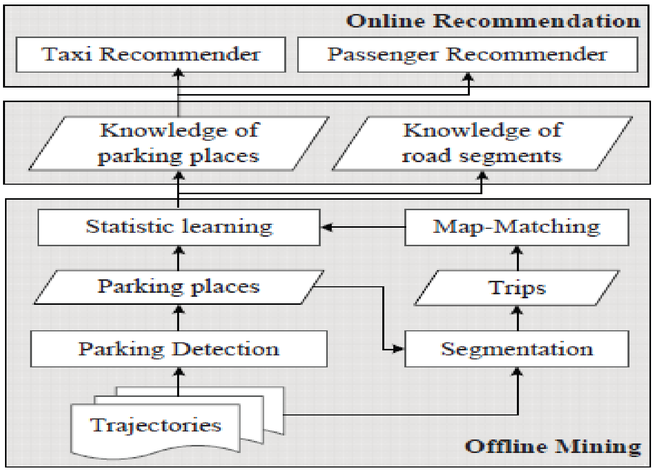

Fig.1 System Overview

5. Model Description

Both the taxi recommender and the passenger recommender are multiple-criteria recommendation systems. In particular, given the current location L_c and time T_0 of a taxi driver or a passenger, the taxi recommender provides the driver with top-*k* parking places and routes to these parking places while the passenger recommender suggests a set of road segments within a walking distance, both according to a specified recommendation strategy, which is either identified by the user's preference or automatically set to a default value.

5.1 Taxi Recommender

The taxi recommender aims to provide the taxi drivers with the best parking places and the routes to these parking places. it's intuitively obvious that a good parking place should bring a high probability (during the routes or at the parking place) to get a passenger, a short waiting time, a short queue length at the parking place and a long distance of the next trip.

5.1.1 Queue Length

In some parking places, taxis often queue to wait for passengers, e.g., at the airports and railway stations. We learn the average queue length at a given time from the historical data. Figure 2 presents the number of arriving vacant taxis at a parking place along the time line. It's clear that the data reveals a certain cyclical characteristics. Since the data in a single day is not enough to get a statistical reasonable result, we first partition the data according to the day of week, and weather condition (normal weather and severe weather), then aggregate the corresponding data into one day (as shown in Figure 3) and estimate the queue length.

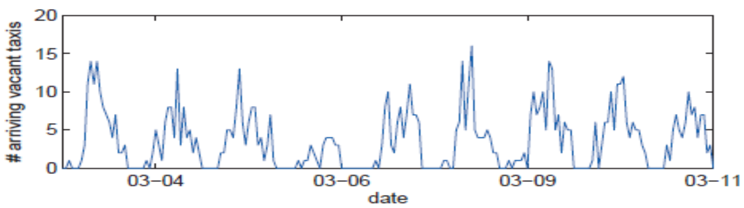

Fig 2: Taxi arriving sequence at a parking place (Queue length estimation)

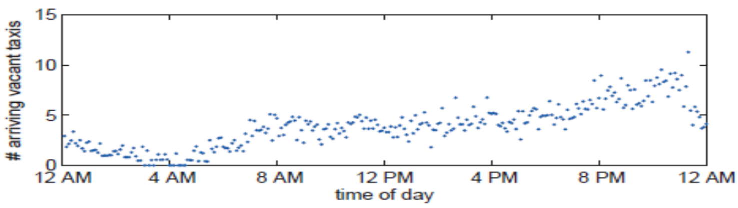

Fig.3: Average number of arriving vacant taxis per day (Queue length estimation)

6. Passenger Recommender

Different from the taxis, the passengers do not want to walk too long for hailing a taxi. If a passenger is close to at least one parking place, we provide him with the nearby parking places with the maximum expected queue length.

6.1 Probability of Finding a Vacant Taxi

Let Pr(*C*; *r*|*t*) be the probability that there is a vacant taxi on road segment *r* at time *t*. Given the passenger's current
position, we suggest him with the road segments, which have the highest probability of finding a cruising taxi among a reachable region Ω of the passenger, i.e.,

$$\begin{aligned} r &= \operatorname*{argmax}_{r\in\Omega} \Pr(\mathcal{C}; r|t) \\ &= \operatorname*{argmax}_{r\in\Omega} \Pr(r|t)\Pr(\mathcal{C}|r,t). \end{aligned}$$

(1)

Let $\{ri\}^R_i$=1 be the set of all the road segments in the road network, then the first factor on the right side of Equation 1

$$\Pr(r|t) = \frac{\sum_{k=\lfloor(t-\Delta t)/\tau\rfloor}^{\lfloor(t+\Delta t)/\tau\rfloor}\left(\#_k(\mathcal{C};r)+\#_k(\mathcal{O};r)\right)}{\sum_{i=1}^{R}\sum_{k=\lfloor(t-\Delta t)/\tau\rfloor}^{\lfloor(t+\Delta t)/\tau\rfloor}\left(\#_k(\mathcal{C};r_i)+\#_k(\mathcal{O};r_i)\right)}$$

(2)

where $\#_k$ (*C*; *r*) and #*k* (*O*; *r*) denote the number of cruising and occupied taxis on road segment *r* within the interval Φ_k respectively. Similarly, the second factor on the right side of Equation 1 is given by:

$$\Pr(\mathcal{C}; r|t) = \frac{\sum_{k=\lfloor(t-\Delta t)/\tau\rfloor}^{\lfloor t+\Delta t/\tau\rfloor}\#_k(\mathcal{C};r)}{\sum_{k=\lfloor(t-\Delta t)/\tau\rfloor}^{\lfloor(t+\Delta t)/\tau\rfloor}\left(\#_k(\mathcal{C};r)+\#_k(\mathcal{O};r)\right)}.$$

(3)

Combining Equation 1, 2 and 3 (note that we only consider the road segments in Ω), we have

$$r = \operatorname*{argmax}_{r\in\Omega} \frac{\sum\limits_{k=\lfloor(t-\Delta t)/\tau\rfloor}^{\lfloor t+\Delta t/\tau\rfloor}\#_k(\mathcal{C};r)}{\sum\limits_{r_i\in\Omega}\sum\limits_{k=\lfloor(t-\Delta t)/\tau\rfloor}^{\lfloor(t+\Delta t)/\tau\rfloor}\left(\#_k(\mathcal{C};r_i)+\#_k(\mathcal{O};r_i)\right)}.$$

7. Validation

7.1 Dataset

Road networks: We can use to evaluate our method using the road network of Beijing, which contains 106,579 road nodes and 141,380 road segments.

Trajectories: The dataset contains the GPS trajectory recorded by over 12,000 taxis in a period of 110 days in the year of 2010. The total distance of the data set is more than 200 million kilometers and the number of points reaches to 577 million.

A sample data is available at [1].

8. Discussion

8.1 Load Balance

The load balance problem is an open challenge for many recommendation systems and is also widely studied in many other fields such as distributed networks[7] and web services[4].

9. Conclusion

To save the time for finding a taxicab and reduce unnecessary traffic flows as well as energy consumptions caused by cruising taxicabs, we proposed a taxi-passenger recommender system based on the pick-up behaviors of high-profit taxi drivers and the mobility patterns of passengers learned from a large number of taxi trajectories. As a result, the taxi recommender accurately predicts the time-varying queue length at parking places and effectively provides the high-profit parking places; the passenger recommender successfully suggests the road segments where users can easily find vacant taxis by our system considering day of the week and weather conditions matches the ground truth for all of the tested areas.

The study provides the following conclusions:

1. We can have a plan to deploy our recommender in the real world so as to further validate and improve the effectiveness and robustness of this system.
2. To save the time for finding a taxicab and reduce unnecessary traffic flows
3. System based on the pick-up behaviors of high-profit taxi drivers and the mobility patterns of passengers learned from a large number of taxi trajectories.
4. Unnecessary energy consumptions caused by cruising taxicabs can be reduced.
5. This system will help to the traffic control officers at the time of traffic jam in metro cities.

References

[1] http://research.microsoft.com/apps/pubs/?id=152883.
[2] J. Yuan, Y. Zheng, C. Zhang, X. Xie, and G. Sun. An interactive-voting based map matching algorithm. In *Proc. MDM 2010*, pages 43–52.
[3] V. Cardellini, M. Colajanni, and P. Yu. Dynamic load balancing on web-server systems. *Internet Computing, IEEE*, 3(3):28–39, 1999.
[4] D. Grosu and A. Chronopoulos. Algorithmic mechanism design for load balancing in distributed systems. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, 34(1):77–84, 2004.
[5] J. Yuan, Y. Zheng, X. Xie, and G. Sun. Driving with knowledge from the physical world. In *Proc. KDD 2011*, pages 316–324. ACM.
[6] C.P. Arora, *Refrigeration and Air conditioning*PHI Publications, 2010.
[7] J. Yuan, Y. Zheng, L. Zhang, X. Xie, and G. Sun. Where to find my next passenger? In *Proceedings of the 13th international conference on Ubiquitous computing*. ACM, 2011.
[8] K. Yamamoto, K. Uesugi, and T. Watanabe. Adaptive routing of cruising taxis by mutual exchange of pathways. In *Knowledge-Based Intelligent Information and Engineering Systems*, pages 559– 566. Springer, 2010.
[9] M. Ziegelmann. *Constrained Shortest Paths and Related Problems*. PhD thesis, Universit¨at des Saarlandes, 2001.
[10] S. Phithakkitnukoon, M. Veloso, C. Bento, A. Biderman, and C. Ratti. Taxi-aware map: Identifying and predicting vacant taxis in the city. In *Proc. AMI 2010*, page 86.