

Efficient Driving Direction Technique Based On Trajectories

S.S.Karthick kumar, G.Indhumathy

Dept of Computer science and Engineering, Velammal College of Engineering and Technology,
Madurai, Tamilnadu, India.

Dept of Computer science and Engineering, Velammal College of Engineering and Technology,
Madurai, Tamilnadu, India.

Abstract---In Efficient driving direction system, the GPS-equipped vehicle are used as mobile sensors thereby probing the traffic rhythm of a city. It helps the drivers in choosing driving directions. The smart driving directions can mine from the historical GPS trajectories. A time-dependent landmark graph is framed with the view of modeling the dynamic traffic pattern and also the intelligence of experienced drivers. Hence it helps assist the user by providing the practically fastest route to a given destination at a given time of departure. The Variance Entropy-Based Clustering method is employed to estimate the distribution of travel time between two landmarks in different time slots. Taking this graph as a basis, a two-stage routing algorithm is used to compute the practically fastest and customized route for end users. Necessarily, the time taken by a driver to traverse a route depends on the following aspects: 1) The physical feature of a route such as distance, capacity (lanes), and the number of traffic lights, number of direction turns; 2) The time-dependent traffic flow along the route; 3) The driving behavior of a user.

Keywords---Driving Directions, Time-Dependent Fast Route, Taxi Trajectories, Landmark Graph

I.INTRODUCTION

GPS-equipped taxis can be considered as mobile sensors to analysis the traffic on road surfaces, and experienced taxi drivers can easily determine the fastest (quickest) route to a given destination based on their knowledge. Smart driving directions will be mine from the historical GPS trajectories of a large number of taxis, and will provide the user with the practically fastest route to a given destination at a given time of departure. The

time-dependent landmark graph, has been proposed where a node (landmark) is a frequently traversed road segment with the view of modeling the intelligence of the taxi drivers and the properties of dynamic road networks. Then, a Variance-Entropy-Based Clustering strategy is applied to estimate the distribution of the travel time between two landmark edges for different time slots. Based on this graph, a two-stage routing algorithm has been designed to compute the practically fastest route. The application such as route planner [7], hot route finder [16], traffic flow analysis [15] have uses the information from Gps data to achieve the better quality of service.

Discovering efficient driving directions has become an essential activity and has been implemented as a key feature in many map service providers like Google and Bing Maps. A fast driving route saves not only the driver's but also the energy consumption (as most gas is wasted in traffic jam). In practice, big cities with serious

traffic problems usually have a large number of taxis traversing on road surfaces. For the sake of management and security, these taxis have already been embedded with a GPS sensor, which enables a taxi to report on its present location to a data center in a certain frequency. Thus, a large number of time-stamped GPS trajectories of taxis have been accumulated and are easy to obtain. Intuitively, taxi drivers are experienced drivers who can usually find out the fastest route to send passengers to a destination based on their knowledge. Apart from the distance of a route, they also consider other factors, such as the time-variant traffic on road surfaces, traffic signals and direction changes contained in a route, as well as the probability of accidents for the selection of driving direction. These factors can be learned by experienced drivers but are too subtle and difficult to incorporate into existing routing engines. Therefore, these historical taxi

trajectories, which imply the intelligence of experienced drivers, provide us with a valuable resource to learn practically fast driving directions.

II. PROBLEM DEFINITION

Intelligence Modeling: As a user is free to select any place as source or destination, there would be no trajectory exactly passing the query points. It is difficult to answer the queries of the user by directly mining the trajectory patterns from the data. Therefore, how to model the intelligence of the taxi drivers that can answer a variety of queries is a challenge

Data Sparseness and Coverage: we cannot guarantee there are sufficient GPS equipped vehicle traversing on each road segment even though there are a large number of vehicles. That is we cannot accurately estimate the speed pattern of each road segment.

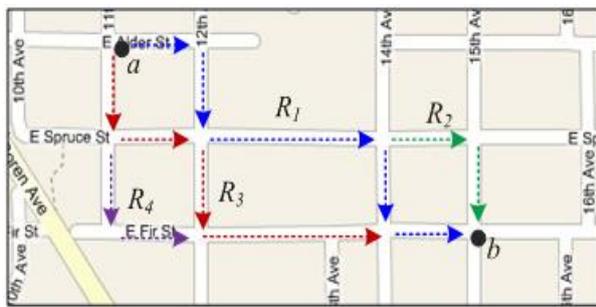


Figure 1: Low Sampling Rate Problem

Low-sampling-rate Problem: GPS equipped vehicle location will be recorded in a very low frequency say 2-5 minutes per point so as to save energy and communication loads [2]. This increases the uncertainty of the roots transverse by taxi [3]. The data obtain from gps trajectories is not precious due to sampling error [17].

III. IMITATIONS OF OTHER TECHNIQUES

Most of the existing mathematics approaches employ the local or incremental algorithms that map current or neighboring positions on to the vector road segments on a map. The current position result will be greatly affected by the measurement errors.

IV. ARCHITECTURE

The architecture of our system consists of three major components: Trajectory Preprocessing, Landmark Graph Construction, and Route Computing. The first two components operate offline and the third is running online. The online parts only need to be performed once unless the trajectory archive is updated. Trajectory Preprocessing: This component first segments GPS trajectories into effective trips, then matches each trip against the road network. 1) Trajectory segmentation:

A tag is associated with a GPS enable vehicles reporting the locations of vehicle

2) Map matching: While dealing with the low-sampling-rate trajectories, and to map each GPS point of a trip to the corresponding road segment the IVMM algorithm, has a better map-matching algorithm than existing map matching algorithm. Landmark Graph Construction: The separate trajectories for the week-days and weekend has been constructed to build the landmark graphs for weekdays and weekends respectively. The two landmarks has been connected with a landmark edge if there are a certain number of trajectories passing these two landmarks. , the distribution of travel time for each landmark edge has been calculated using the VE-clustering algorithm.

Route Computing: For a given a query (qs, qd, td), two-stage routing algorithm will be carried to find out the fastest route. In the first stage, rough routing will searches the time-dependent landmark graph for the fastest rough route represented by a sequence of landmarks. In the second stage, the refined routing algorithm will be employed to compute a detailed route in the real road network to sequentially connect the landmarks in the rough route.

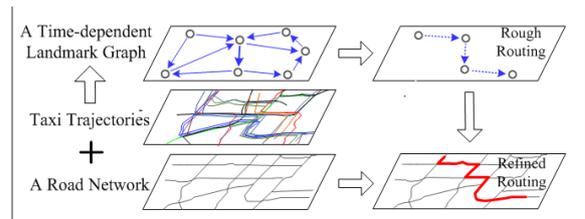


Figure 2: Hierarchical Architecture

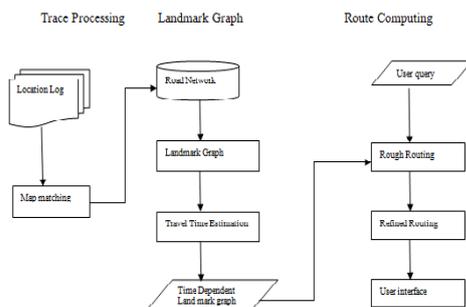


Figure 3: System Architecture

The trajectories has been build for that different week-days (e.g. Tuesday and Thursday) almost share similar kinds of traffic patterns while the weekdays and weekends have different patterns. Therefore two different landmark graphs are built for weekdays and weekends respectively. All the weekday trajectories (from different weeks and months) into one weekday landmark graph, and put all the weekend trajectories into the weekend landmark graph. The threshold δ is used to eliminate the edges that are seldom traversed by taxis, because, fewer the taxis that pass two land-marks, lower the accuracy of the estimated travel time (between the two landmarks). Additionally, the t_{max} value used to remove the landmark edges has a very long travel time. Due to the low-sampling-rate problem, sometimes, a taxi

may consecutively traverse three landmarks while no point is recorded while passing the middle (second) one. This will result the travel time between the first and third landmark being very long. Such types of edges will increase the space complexity of a landmark graph and results inaccuracy to the travel time estimation.

A. Trajectory Preprocessing:

This component first segments GPS trajectories into effective trips, then matches each trip against the road network. 1) Trajectory segmentation: In practice, a GPS log may record a taxi's movement of several days, in which the taxi could send multiple passengers to a variety of destinations. Therefore, we partition a GPS log into some taxi trajectories representing individual trips according to the taximeter's transaction records.

B. Landmark

A landmark is one of the top-k road segments that are frequently traversed by taxi drivers according to the trajectory archive. The reason for using the "landmark" for modeling the taxi drivers' intelligence is: 1) the notion of landmarks follows the natural thinking pattern of people, and can give the user with more understandable and memorable presentation of driving directions beyond detailed descriptions. For instance, the typical pattern that people introduce a route to a driver is like this take I-406

East at NE 5th Street, then change to I-90 at exit 11, and finally exit at Qwest Field". Instead of giving turn-by-turn directions, people prefer to use a few landmarks (like NE 5th Street) that highlight key directions to the destination.

2) The low sampling-rate and sparseness of the taxi trajectories do not support the speed estimation for each road segment while the traveling time between two landmarks can be estimated. Meanwhile, the low-sampling-rate trajectories cannot offer sufficient information for inferring the exact route here, top-k road segments can be detect as the landmarks instead of setting up a fixed threshold.

familiarity of routes. For example, people who are familiar with a route can usually pass the route faster than a new-comer. Sometimes, even on the same path, cautious people are likely to drive relatively slower than those preferring to drive quickly and aggressively.

E. Clustering

The number of clusters and the boundary of these clusters vary in different landmark edges, therefore the following V-Clustering instead of using some k-means-like algorithm or a predefined partition. V-Clustering: VE clustering [1] is an two phase clustering method to learn different time partitions for different land mark edges based on trajectories. We first sort Tuv according to the values of travel time (tl- ta), and then partition the sorted list L into several sub-lists in a binary-recursive way. In each iteration, compute the variance of all the travel times in L. Later, find out the best split point having the minimal weighted average variance (WAV) defined

$$WAV(i;L) = |L1(i)|/|L|Var(L1(i)) + |L2(i)|/|L|$$

Algorithm 1: Landmark Graph Construction

Input: a road network Gr, a collection of road segment Sequences M, the number of landmarks k, the Thresholds and tmax

Output: landmark graph Gl

```

1 M <- E <- ∅, Count[ ] 0;
2 foreach road segment sequence S ∈ M do
3   foreach road segment r ∈ S do
4     Count[r]++
5   V1 <- Top-k(Count[ ], k);
6   foreach road segment sequence S ∈ M do
7     S <- Convert(S,V1)/* Converted to landmark
sequence */
8   for i <- 1 to |S|-1 do
9     u <- S[i], v <- S[i + 1] ;
10    if v.t - u.t < tmax then
11    if euv = (u, v; Tuv) ∈ E then
12    E.Insert(euv)
13    euv:supp++;
14    Tuv.Add((u,t; v,t))
15    foreach edge e = (u,v; Tuv) ∈ E do
16    if e:freq < δ then E.Remove(e) ;
17    return Gl <- (V1,E);
    
```

C. Transition

Given a trajectory archive A, a time threshold tmax, two landmarks u; v, arriving time ta, leaving time tl, we say s = (u; v; ta; tl) is a transition if the following conditions are satisfied:

(I) There exists a trajectory Tr = (p1, p2, ..., pn) ∈ A, after map matching, Tr is mapped to a road segment sequence (r1, r2, ..., rn). E I, j, 1 < i < j < n s.t. u = ri; v = rj .

(II) ri+1, ri+2, ..., rj-1 are not landmarks.

(III) ta = pi.t, tl = pj.t and the travel time of this transition is tl- ta < tmax.

D. Rough Routing

Besides the traffic condition of roads, the travel time along a route depends on the behavior of the drivers. Sometimes, different drivers take different amount of time to traverse the same route at the same time. The reasons depend on the driver's driving habit, skill and

Var(L2(i)) where L1(i) and L2(i) are two sub-lists of L split at the ith element and Var represents the variance. This best split point leads to a maximum decrease of

$$\Delta V(i) = Var(L) - WAV(i;L)$$

Algorithm 2: Variance-Entropy-Based Clustering

Input: a set of points S = {(xi; yi)ni=1} ⊆ R*R

Output: a sequence of distributions D1; D2, ..., Dk

```

1 Sy <- sorted sequence {yi}ni=1 1 order by yi
ascending;
2 y_split <- ∅;
3 y_split <- V-Clustering(Sy, δ v, y_split);
4 C = {c1, c2, ..., cm} <- Convert(Sy, y_split);
/* Convert Sy into clusters according to y_split */
5 Sxc <- sort {(xi; c(yi))ni=1} order by xi ascending;
/* c(yi) ∈ C is the cluster of yi */
6 x_split <- ∅;
7 x_split <- E-Clustering(Sxc, _e, x_split);
/* Divide x-axis into several slots */
8 for i <- 1 to |x_split| do
9   Di <- Compute Distribution(Sxc, i, x_split);
/* Compute the distribution of slot i */
    
```

10 return D = {D1;D2.....,Dk};

V. CONCLUSION AND FUTURE WORKS

This approach finds out the practically fastest route to a destination at a given departure time in terms of taxi drivers' intelligence learned from a large number of historical taxi trajectories. First step to construct a time-dependent landmark graph, and then perform a two-stage routing algorithm based on this graph to find the fastest route. Significantly outperforms both the speed-constraint-based and the real-time traffic-based method in the aspects of effectiveness and efficiency. However there exist some problems that a recommended route would become crowded if many people take it.

This is the common problem of Path-finding, and this problem is even worse (than ours) in present shortest-path and real-time-traffic-based methods (as our method can be customized for different drivers). In the future, this can be solved by using some strategies, such as load balance and data update

REFERENCES

- [1] R. Chhikara And L. Folks. The Inverse Gaussian Distribution: Theory, Methodology, And Applications. 1989.
- [2] K. Cooke And E. Halsey. The Shortest Route Through A Network With Time-Dependent Internodal Transit Times. *J.Math. Anal. Appl.*,14(492-498):78.
- [3] B. C. Dean. Continuous-Time Dynamic Shortest Path Algorithms. Master's Thesis, Massachusetts Institute Of Technology, 1999.
- [4] B. Ding, J. Yu, And L. Qin. Finding Time-Dependent Shortest Paths Over Large Graphs. In *Proc. Edbt*, Pages 205-216. Acm, 2008.
- [5] S.DREYFUS. AN APPRAISAL OF SOME SHORTEST-PATH ALGORITHMS. *OPERATIONS RESEARCH*, 17(3).
- [6] FAYYAD AND IRANI. MULTI-INTERVAL DISCRETIZATION OF CONTINUOUS-VALUED ATTRIBUTES FOR CLASSIFICATION LEARNING. *PROC. IJCAI*, PAGES 1022-1027, 1993.
- [7] Gonzalez, H., Han, J., Li, X., Myslinska, M., And Sondag, J. P. 2007. Adaptive Fastest Path Computation On A Road Network: A Traffic Mining Approach. In *Proceedings Of The 33rd International Conference On Very Large Data Bases* 794-805, 2007.
- [8] Y. ZHENG, Q. LI, Y. CHEN, X. XIE, AND W. MA. UNDERSTANDING MOBILITY BASED ON GPS DATA. IN *PROC. UBIComp*, PAGES 312
- [9] E. KANOULAS, Y. DU, T. XIA, AND D. ZHANG. FINDING FASTEST PATHS ON A ROAD NETWORK WITH SPEED PATTERNS. IN *PROC. ICDE*, 2006.
- [10] Y. ZHENG, L. LIU, L. WANG, AND X. XIE. LEARNING TRANSPORTATION MODE FROM RAW GPS DATA FOR GEOGRAPHIC APPLICATIONS ON THE WEB. IN *PROC. WWW*, PAGES 247-256, 2008
- [11] Y. LOU, C. ZHANG, Y. ZHENG, X. XIE, W. WANG, AND Y. HUANG. MAP-MATCHING FOR LOW-SAMPLING-RATE GPS TRAJECTORIES. IN *PROC. GIS. ACM*, 2009.
- [12] A. ORDA AND R. ROM. SHORTEST-PATH AND MINIMUM-DELAY ALGORITHMS IN NETWORKS WITH TIME-DEPENDENT EDGE-LENGTH. *JACM*, 37(3):625, 1990.
- [13] D. PFOSE, S. BRAKATSOULAS, P. BROSCHE, M. UMLAUFT, N. TRYFONA, AND G. TSIRONIS. DYNAMIC TRAVEL TIME PROVISION FOR ROAD NETWORKS. IN *PROC. GIS. ACM*, 2008.
- [14] J. YUAN, Y. ZHENG, C. ZHANG, AND X. XIE. AN INTERACTIVE-VOTING BASED MAP MATCHING ALGORITHM. IN *PROC. MDM*, 2010.
- [15] Kuehne, R. Et Al. New Approaches For Traffic Management In Metropolitan Areas. In *10th Ifac Symposium On Control In Transportation Systems*, 200
- [16] Li, X., Han, J., Lee, J., And Gonzalez, H. Traffic Density-Based Discovery Of Hot Routes In Road Networks. In *10th International Symposium On Spatial And Temporal Databases*, 2007
- [17] Pfose, D. And Jensen, C. S. Capturing The Uncertainty Of Moving-Object Representations. In *Proceedings Of The 6th International Symposium On Advances In Spatial Databases*, 111-132, 1999.