Inferring Visiting Time Distributions of Locations from Incomplete Check-in Data

Hsun-Ping Hsieh
Graduate Institute of Networking and Multimedia,
National Taiwan University, Taipei 106, Taiwan
d98944006@csie.ntu.edu.tw

Cheng-Te Li
Graduate Institute of Networking and Multimedia,
National Taiwan University, Taipei 106, Taiwan
d98944005@csie.ntu.edu.tw

ABSTRACT

Online location-based services, such as Foursquare and Facebook, provide a great resource for location recommendation. As we know the time is one of the important factors on recommending places with proper time for users, since the pleasure of visiting a place could be diminished if arriving at wrong time, we propose to infer the visiting time distributions of locations. We assume the check-in data used is incomplete because in real-world scenarios it is hard or unavailable to collect all the temporal information of locations and the check-in behaviors might be abnormal. To tackle such problem, we devise a visiting time inference framework, *VisTime-Miner*, which considers the route-based visiting correlation of time labels to model the visiting behavior of a location. Experiments on a large-scaled Gowalla check-in data show a promising result.

Categories and Subject Descriptors

H.2.8 [**Database Management**]: Database Applications–*Data mining*.

Keywords

Location visiting time, check-in data, semi-supervised.

1. INTRODUCTION

The pleasure of visiting a place can be significantly diminished if arrived at the wrong time. Some places have a wider range of visiting time span while others are constrained to certain particular time slots. For example, most people do not want to visit a beach during the boiling hot noon, but rather arrive in the late afternoon to enjoy the sunset scene. Or certain ball game events usually take place at particular time period (e.g. in the evening). As shown in Figure 1, which is derived from the Gowalla check-in data, some place has better chance to be visited at certain time slots. For example, people visited Empire State Building from about 12:00 to the mid night (note that this place is famous for its excellent night view), in contrast, the proper time to visit Central Park is during daytime.

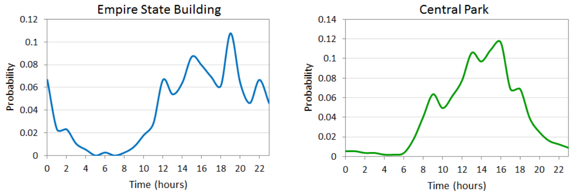

Figure 1: The distribution of the visiting probability at each time unit (hour) for Empire State Building (left) and Central Park (right).

In this paper, given a particular location, we aim to infer its visiting time distribution from incomplete check-in data. Acquiring the visiting time distributions of locations would enable a number of applications. For example, one can provide time-aware location recommendation [4] based on the time at a current location. Besides, the quality of route planning [3] can be boosted if the visiting time of locations can be derived. Moreover, while existing work models the periodic and social mobility [1] for location prediction, it would be more realistic if the proper location visiting is considered. However, as inferring the location visiting time from real-world location-based services, we encounter two critical challenges on obtaining enough and accurate time-labeled check-in data. First, the number of new locations or point-of-interests (POI) is rapidly generated due to new events, buildings, attractions, even new developed areas, and so on. It is hard and infeasible to acquire the all the complete visiting information of locations. What we have could be incomplete or few records on the new locations. Second, even though we have lots of check-in records of locations, such data has high potential to contain noise and abnormal check-in behaviors. We might have to data correction or manually annotate the data before applications. Such processes are expensive and time-consuming. Therefore, in this paper, our goal lies in inferring the visiting time distribution of locations from incomplete check-in data. We assume the data used is partial time-labeled and only a small amount. That is, we have only limited time-labeled information while most of locations are unlabeled.

We propose a novel inference framework, *VisTime-Miner*, to infer the visiting time distributions (*VTD*) of locations. The *VTD* of a location is a probability distribution at each time unit (hour), such as Figure 1. Since we use very few time-labeled data (less than 1%) on each location in a city, our method is built on a semi-supervised inference model. The central idea is two-fold: (a) the visiting order of locations in a route reveals their time information, and (b) locations are temporally similar if they are also similar on geographical and contextual aspects. Experimental results show that our method is promising as significantly outperforming reasonable baselines.

Related Works. Some recent work exploits complete time-labeled check-in data for location recommendation. Yuan et al. develop a collaborative recommendation model [4] to recommend POIs for a given user at a specified time in a day. Hsieh et al. develop a *TripRouter* system [3] to construct time-sensitive routes, which consider location popularity, visiting order, proper visiting time, and proper transit time to model the goodness of a route. Wei et al. [6] develop a route inference to construct the popular routes passing through a given location sequence within a specified time span. Sadilek et al. [5] predict the most likely location of an individual at any time, given the historical trajectories of his/her friends. To the best of our knowledge, we are the first to tackle the visiting time inference problem using check-in data.

2. THE PROPOSED METHOD

Notation. A route is a sequence of time-labeled locations, denoted by $r = <(l_1, t_1), (l_2, t_2), ..., (l_k, t_k)>$, in which l_i is a location and t_i is the corresponding time label in hour (i.e., 0, 1, 2, ..., 23). A time-labeled route is a route, in which each location is associated with a time stamp in hour. A time-unlabeled route is a route, in which all the locations have no time stamps associated. A visiting time distribution (*VTD*) of location l_i is a probability distribution over time labels in hour, denoted by $VTD(l_i) = <(t_0, p_0), (t_1, p_1), (t_2, p_2), ...,$

WWW'14 Companion, April 7–11, 2014, Seoul, Korea.
ACM 978-1-4503-2745-9/14/04.
http://dx.doi.org/10.1145/2567948.2577362

(t_{23}, p_{23})>, where $p_0 + p_1 + ... + p_{23} = 1$. Given a set of routes, we define the time-label ratio (denoted by τ) as the number of time-labeled routes divided by the number of time-unlabeled routes. Note that τ is usually less than 5% and is varied for the evaluation.

Problem Definition. Given a set of routes with a certain time-label ratio, our goal is to infer the $VTD(l_i)$ for each location l_i in the routes. In other words, we aim to infer the visiting time for the locations in the time-unlabeled routes. It is because of that if we can obtain all the visiting time of locations in unlabeled routes, we then can aggregate and derive their *VTD*.

[Step 1] Construct Route-Correlated Graph. For location l_i that appears in n_i routes (including both labeled and unlabeled), we construct a weighted *complete* graph (*RCG*) with n_i nodes, in which some nodes have time labels (i.e., those come from time-labeled routes) while others do not. We compute edge weights considering the similarity between routes. The fundamental idea is that if two routes are more similar, their time labels of location l_i tends to be more correlated. Therefore, we give higher edge weights if two routes are more similar. Here we propose a novel route similarity, which consists of three parts: (a) *location overlapping* is the Jaccard coefficient on the location sets of two routes. If two routes have more locations overlapped, they will get higher score. (b) *position difference* is the reciprocal of the maximum position difference of location l_i between two routes, smoothed by adding one. If a location is visited at a relatively-close position on two routes, their edge weight gets higher. (c) *geographical proximity* is the average distance in geography over locations between two routes. After some normalization, we calculate the geometric mean of such three scores and regard the value as the edge weight between two routes of location l_i in *RCG*.

[Step 2] Learn the visiting time label. We learn the visiting time label for each location in time-unlabeled routes, by leveraging the graph-based semi-supervised inference technique [2], which exploits Gaussian random fields and harmonic functions to relax the Boltzmann machines. The basic idea is to optimize the loss function based on the constructed route-correlated graph such that the labeled data are clamped. Since the inference process goes beyond the scope of ours, please refer to Zhu et al.'s work [2] for details.

[Step 3] Infer the probability with the corresponding visiting time label for each location in an unlabeled route. From Step 2, we derive the time label distribution over time labels for each location l_i in each time-unlabeled route. We would like to further consider the visiting order of locations in a route to infer the most proper probability of l_i. The idea is that users usually tend to visit locations along a route with the most proper time. Therefore, given a time-unlabeled route with location sequence <$l_1, l_2, ..., l_k$>, we aim to find the corresponding time label probabilities $p_1, p_2, ..., p_k$ such that $\prod_{i=1..k} p_i$ is maximized, under the constraint that the visiting time of the $(i\text{-}1)^{th}$ location should not be later than the visiting time of i^{th} one. We adopt dynamic programming technique to find such probabilities.

[Step 4] Aggregate the probabilities to derive VTDs. Finally, we aggregate the probabilities with the corresponding visiting time labels by simple statistic counting. Then the predicted visiting time distributions for locations in each time-unlabeled route are derived.

3. EXPERIMENTS

We use a large-scaled check-in data from Gowalla [1] for the evaluation, which contains 6,442,890 check-in records. By constraining a route of a user within a day, we obtain 1,136,737 routes. We extract two check-in subsets falling into the urban areas of New York and San Francisco. By varying the time-label ratio, we randomly select the time-labeled routes while the time labels of the remaining routes are removed. We aim to infer the *VTD*s of all the unlabeled locations in each city. We use the *symmetric Kullback-Leibler* (*KL*) *Divergence* as the evaluation metric, which measures the difference between the ground-truth *VTD* and the inferred *VTD*, the smaller the better. The average *KL* Divergence value over all unlabeled locations in a city will be reported. For KL, we develop two baseline methods: (a) *normal distribution* method simply generates a Gaussian distribution whose mean locates at the most frequent check-in time (in hour) in the labeled data. (b) We construct the *VTD* from the *labeled data*, which serves as a strong baseline. The other evaluation measure is *hit rate*, defined as the number of successfully predicted check-in locations divided by the number of unlabeled check-in locations over all time-unlabeled routes. Higher hit rate indicates better quality of inference. For hit rate, the comparative method always chooses the most frequent check-in time label as the predicted one from the time-labeled routes. Figure 2 and Figure 3 shows the results. We can find our *VisTime-Miner* significantly outperforms the baseline methods.

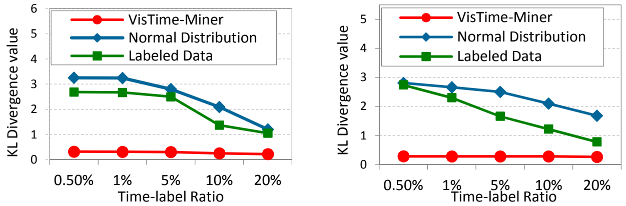

Figure 2: Average KL divergence values for New York (left) and San Francisco (right), by varying the time-label ratio.

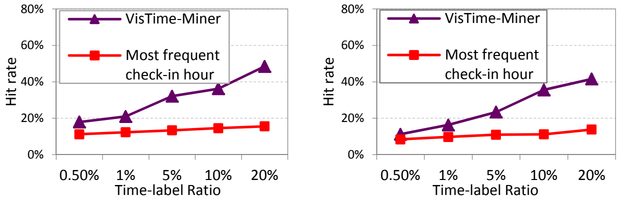

Figure 3: Hit rate for New York (left) and San Francisco (right), by varying the time-label ratio.

4. Conclusion

This paper proposes to infer the visiting time distributions of locations from incomplete check-in data. We consider the route-based visiting correlation of time labels to model the visiting behavior of a location, and devise the semi-supervised learning framework, *VisTime-Miner*, to tackle the visiting time inference problem. Experiments on Gowalla check-in data show the promising results. Ongoing work is to use the location visiting time for time-aware applications such as route planning and recommendation.

5. REFERENCES

[1] E. Cho, S. A. Myers, and J. Leskovec. Friendship and mobility: User movement in location-based social networks. In *ACM KDD* 2011.

[2] X. Zhu, Z. Ghahramani and J. Lafferty. Semi-Supervised Learning Using Gaussian Fields and Harmonic Functions. In *ICML* 2003.

[3] H.-P. Hsieh, C.-T. Li and S.-D. Lin, Measuring and Recommending Time-Sensitive Routes from Location-based Data. In *ACM TIST* 2014.

[4] Q. Yuan, G. Cong, Z. Ma, A., Sun and N. M. Thalmann. Time-aware point-of-interest recommendation. In *ACM SIGIR* 2013.

[5] A. Sadilek, H. Kautz, and J. P. Bigham. Finding your friends and following them to where you are. In *ACM WSDM* 2012.

[6] L.-Y. Wei, Y. Zheng, and W.-C. Peng. Constructing Popular Routes from Uncertain Trajectories. In *ACM KDD* 2012.