Learning and User Adaptation in Location Forecasting

Jorge Alvarez-Lozano
Computer Science Department,
CICESE Research Center
Carr. Ensenada-Tijuana 3918
Ensenada, México
jalvarez@cicese.edu.mx

J. Antonio García-Macías
Computer Science Department,
CICESE Research Center
Carr. Ensenada-Tijuana 3918
Ensenada, México
jagm@cicese.mx

Edgar Chávez
Institute of Mathematics,
UNAM
México
elchavez@gmail.com

http://dx.doi.org/10.1145/2494091.2495978

Abstract
User location forecasting is central to establish context in proactive mobile applications. Knowing where the user will be at a given time enables standby action triggers ahead of time. User location exhibits periodic patterns grouped by time of day, day of the week, month of the year, etc. This characteristic has been exploited to model user location as a Markov process with great accuracy. Using yearly data from public sources it was possible to predict user location in a time frame of 8 hours with accuracy of up to 69 %.

One assumption of the above modeling is that user location is stationary in time. However, it is more natural to assume user location patterns may vary over time. For example one user may change job, or the relationship status, and avoid certain places frecuented in the past.

In this paper we propose a learning mechanism adapting user location forecasting to behavior changes over time. Our model is able to predict for up to 94 weeks with 43 % of accuracy.

Author Keywords
User location forecasting, Updating user location model, User mobility patterns

ACM Classification Keywords
G.3 [Mathematics of Computing]: Markov processes; G.3 [Mathematics of Computing]: Reliability and life testing

Introduction
Mobile devices have become an ever present garment carried by users at all times and all places. Mobile devices represent a very rich source of contextual data, in particular they can establish the location of a user at any time. Passive location detection (reading the current location) is useful for reactive applications where an action is triggered once the user is in place. Proactive applications need to know where the user will be at a given time, among a selection of discrete places or regions. In the literature we find some research stating that people exhibit a high degree of repetition visiting regular places during their daily activities [4]. This aspect has been exploited to forecast the next location of the user. Meanwhile, other works have defined methods to predict the user location in space and time (*spatio-temporal prediction*) for short and long-term time lapses [15, 14, 1] with varied success, as discussed in the related work.

For a proactive mobile application location is discrete (for example, arriving to work, home, a restaurant, etc.). We will assume the location event happened or not. This will be important to our modeling. We will call this discrete locations, *Points of Interest* (POI). Our goal will be to know where a user will be at a given time, or when the user will be at a given POI. To fix ideas, lets see one example of a proactive application using the location: assume the application has a considerable user base and without user intervention the application forecasts the user location in the next hour. One particular user, queries the application about the capacity of a given POI (e.g. a restaurant) in a few hours. Using the data from all the users, the application can forecast the capacity of the restaurant and alert the user about this.

The above setup has been explored in previous work, however we noticed a gap between our assumptions and a more realistic setup. The assumption is that users will have the same mobility patterns along time. This is true to some landmarks like home and work, as it is not likely that a user will change work or home in short periods. For other POI like restaurants, movie theaters, etc. users are likely to be adventurous and may change preferences over time. In this paper we tackle this problem with two questions: How can we learn user mobility changes over time?, or in other words: How can we adapt our algorithm to changes in user mobility patterns?

Up to the best of our knowledge this is the first attempt to include explicit changes in mobility when forecasting user location. Previous work, discussed below, assumes user mobility patterns are stationary. Since we have different hypothesis we expect a smaller prediction rate than previous work.

Related work
Previous work assume a stationary behavior in user mobility. We can distinguish the related work as *spatial and spatio-temporal prediction*. Spatial prediction identifies the next place or places where the user will be, but not the time when the user will be at the place [2, 6, 7]. In [7], authors forecast the driver's destination using GPS trajectories as the journey progresses and the driver moves from a geographical region to another. In [18] the aim is to predict in which cell of a cellular network the user will be, being the restriction the topology, there are only a limited number of cells the user can reach from the current location. In WhereNext [10], authors use

previous trajectory patterns, which are modeled as a tree. Then, considering a new trajectory the next location is predicted finding the most likely sequence path in the tree. A more recent work uses Hidden Markov Models (HMM), and the same dataset as us. However, they define on a different way the HMM; they are interested in the mobility among geographical regions instead of POIs [9].

The aim of the spatio-temporal prediction is to forecast user location in space and time. In [17] traces of Wi-Fi and Bluetooth are used to find regular patterns in order to predict where a user will be at a given time, how long the user will stay at that place, and also infer who the user will meet there. In [1] we presented a spatio-temporal prediction model using discrete location data, restricting the scope to places with internet connectivity. In NextPlace [15], the authors get as reference the sequence of the current places visited by a user, and using the historical records search for a similar pattern in the past to predict the next place the user will visit and the time the user will stay there. Although the above methods have been useful in some domains, they do not include the mobility change over time to forecast user location. Also, these work do not have a good accuracy when a medium-term prediction time is considered (some hours), and do not distinguish a particular pattern for each day of the week. Up to the best of our knowledge there is no work using an updating or correction mechanism.

User mobility

Although user mobility seems to be dynamic, most people follow certain patterns [4, 3]. Mobility is defined by our daily activities and habits, such as going to work, and other activities that vary over time. We can distinguish between weekday, weekend, monthly or annually patterns [11, 3, 18]. Once recognized the user mobility patterns, it is possible to predict user location in space and time. In this paper we fixed our attention in three characteristics of user mobility and considering their interrelation we can infer the user mobility over time.

We can assume certain periodicity in location/time user patterns. Usually weekdays are similar, people tend to organize their life according to work or school hours, and our hypothesis is that activities in the same weekday will have a repetitive pattern. A corresponding periodicity is observed during weekends. Mobility exhibits a different pattern for each day of the week [11] (*first characteristic*). Hence, the places visited in a given day are postulated to be the same for the subsequent days. We also postulate that current location in a given day and hour conditions the next place to be visited (*second and third characteristic* respectively). For example if one user is at home at 7:00 AM on a Monday, the next place he will be at is most likely the coffee shop or the office, but not the movie theater. We postulate that user mobility is a Markovian stochastic process and can be described with a Markov Chain. The Markovian property [8] states that current place is only a function of the previous place. Our main claim is that once the data is grouped by repetitive patterns (day and time over a time window), the sequence of places visited form a Markovian chain.

We are aware that this claim is just valid when user moves among certain discrete places. For example in a traffic detour due to an unscheduled maintenance to the highway, no model will be able to catch the next location of a user facing detours. These discrete locations are distinguished because the user spends some time there and visit them frequently, this will be a characteristic of a POI. Also, we can state that the mobility among POIs does not change; just the places visited in the transition among POIs.

Taking into account the user mobility characteristics, user mobility can be modeled as an HMM [12] (Figure 2). A hidden Markov model is a finite state machine consisting of a set of hidden states (Q), a set of observations (O), transition probabilities (A), emission probabilities (B), and initial probabilities for each state (π). Hidden states are not directly visible, while observations (dependents on the state) are visible. Each state has a probability distribution over the set of observations. In our case, the hidden states corresponds to POIs, which have a probability distribution over times of day. Once the HMM has been defined, three problems can be addressed, (i) finding the probability of an observed sequence (evaluation); (ii) finding the sequence of hidden states that most probably generated an observed sequence (decoding); (iii) generating a HMM given a sequence of observations (learning). For the purpose of this work, we use the *decoding approach*; given a time period (seq. of observations), we want to know the most likely sequence of locations where the user will be (hidden states).

The proposed approach

Some steps are required to forecast user location over time. First, we identify POIs using location data correspondent to a sliding window. Then we define an HMM considering user mobility among these POIs. Once the HMM is defined, we make some predictions. Finally, the prediction model is updated using the most recent mobility data. Please notice that identifying points of interest is an important task per se.

Identifying points of interest

Nowadays, mobile devices provide continuous and discrete location data (e.g. GPS records,WiFi records, or presence status). Thereby, considering the great amount of location data there is a need for algorithms that deal with the challenge of turning data into significant places [2, 5, 13, 6]. These works can be categorized as: residence time-based [15, 6, 5], density-based clustering [13], and lost signal-based [2]. As we want to consider the current mobility to define a better prediction model, it is necessary to merge all the information sources in a coherent way. We need to identify significant places and rule out outdated data, or non significant mobility behaviors. Situations we need to be aware of include when a user was at a given place for a long time, months ago; or, considering the density-based approach, a place can be considered significant after several visits. Therefore, it is necessary to consider some variables: time and location data, frequency of visits, residence time, cluster radius (size of POI), and a sliding time window (which covers the current mobility behavior). The cluster radius govern the granularity we will have in identifying POIs (e.g. home will be smaller than a shopping mall).

Ashbrook and Starner's algorithm [2], focuses on discovering indoor POIs. To do that, they consider the loss of GPS signal within a radius, and a time threshold for the disappearing period. Kang *et al* [5], propose a time-based clustering algorithm to discover outdoor POIs. They compare an incoming GPS reading with previous readings in a current cluster; if the stream of reading moves away from the current cluster then they form a new one. If the GPS readings are close together (within some distance d of each other), and the user spends at least t minutes at that region, a cluster is formed; the cluster is discarded if the user stays less than t time. For residence time, we used the value suggested by [2](10 minutes), and for the cluster radius we used three values (100, 250, and 500 meters); for the frequency, a place must have at least n visits, where $n =$ sliding window size/2.

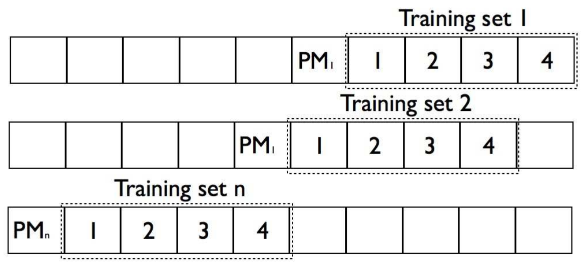

Figure 1: Using a sliding window to update the prediction model.

Updating POI

As user mobility varies over time we want to know the performance of our spatio-temporal prediction model after including mobility changes. For example, a given user is starting guitar classes on Monday, from 7:30 AM to 8:30 AM at a music school. For the subsequent Mondays, the user takes his classes at the same time; however, the music school is not a POI yet because it does not have the number of visits required. Some weeks after, the music school has the required visits to be a POI; the HMM is updated. The HMM also is updated when a place ceases to be a POI. To address this aspect we propose to use a time sliding window taking into account the most recent mobility behavior. Considering Figure 1, the first prediction model (PM_1) will be set using the records of the last 4 weeks (*Training set 1*), then the sliding window moves; PM_2 will be set with the records of the second training set, and so forth.

Defining spatio-temporal prediction model

The prediction model (Figure 2) is defined as follows:
Hidden states are defined by the POIs. Another state was added to define when the user is not at a POI.
Observations are defined by different slots (covering time periods of 2, 3, and 4 hours).

Vector π defines the probability that the user starts his day at a given POI.
Transition matrix defines the probability that the user moves from a POI to another, or from a POI to the state that corresponds to a non-POI.
Confusion matrix defines the probability that the user is at a given POI at a given slot (time period).

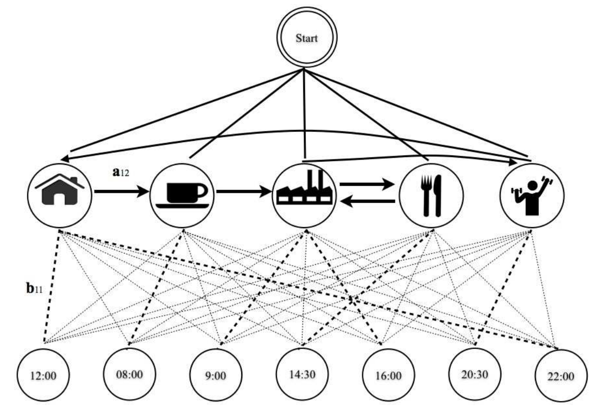

Figure 2: An HMM representing some POIs and their relationship with different hours of the day.

Once the HMM is defined, we are able to forecast user location in a given time period. For instance, if the current time is 11:30 AM, and we want to know where the user will be in the period 11:30 - 19:30, the HMM parameters are used to identify the combinations of hidden states (and their corresponding probabilities) that satisfy such time period, so later the sequence of hidden states with the highest probability can be selected (*decoding problem*). In order to identify the most likely sequence of hidden states we use the Viterbi's algorithm [16]. This is a trellis algorithm that finds the single best state sequence given an observation sequence (Figure 3).

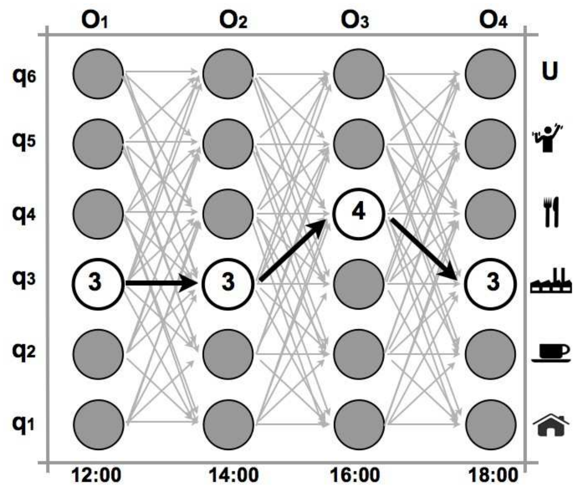

Figure 3: Using Viterbi's algorithm to identify the sequence of POIs where the user will be in a given time period.

Evaluation

To know the accuracy of our prediction model, we did several tests using different configurations for the HMM. We used the GeoLife dataset [19, 20]. This dataset contains GPS trajectories of 178 users collected in a period of over 4 years from April 2007 to October 2011. A GPS trajectory is represented by a sequence of time-stamped points, each of them contains the information of latitude and longitude. Although the dataset has been collected during a period of 4 years, there is not a great amount of location data for each individual user. After analyzing the records for each user, we decided to choose 18 users; who have GPS records over several months (Table 1). This aspect allows us to know the performance of our prediction model after updating it over time. For each user, we grouped the records according to the day when they were created. This way, we use the historical records of a given day of the week to define the prediction model for this specific day; seven prediction models were defined in order to characterize user mobility by day. For each prediction model, we used different configurations: cluster radius (100, 250, 500 m), sliding window, and different values for observations. We used two values for the sliding window: 4 and 8 weeks. Thus, we can know the variation of number of POIs, and the accuracy of the prediction model considering location data of different time periods. Regarding to observations, the day has been split in several slots considering different values (2, 3, and 4 hours). According to the predictions, for each model we made 4 predictions considering different values for ΔT (1, 2, 4, and 8 hours); 28 predictions for each user and each week tested. Each prediction is uniformly distributed over the [00:00,10:00] interval. We selected this interval because most of the GPS trajectories are in the interval [00:00,18:00], thereby if $T = 10:00$, it is possible to know in which POIs a user will be using the largest value for ΔT. In total, 5186 predictions were made using a sliding window of 4 weeks, and 2872 when we considered a sliding window of 8 weeks. All prediction models were defined using a $1st$ order HMM. Finally, the effectiveness is defined as: for a prediction interval $[T, T + \Delta T]$, the prediction is correct if the user is at POI q_i at the time period $T_{pred}(T_{pred} = T + o_i \quad 1 \leq i \leq$ No. of obs. in the interval). That is, a prediction is correct when the user is at the POI defined by q_i, at the time period indicated by the observation o_i. It would also be correct if the prediction indicates that the user is not at a POI (in the case of the state corresponding to an unknown place).

Results

Variation of number of POIs

Using different cluster radius size leads to get a variation in the number of POIs, and the amount of time that the user spends at them. For example, Figures 4 and 5 present the time distribution for a user with ID=2 (for lack of space we omit the distribution for a cluster of 250 m.). Considering these distributions, it can be seen that for this user there is a significant variation in the number of POIs and time distribution; the number of POIs decreases from 25 to 12, and considering a 500 m. cluster radius most of the residence time for this user was spent at a couple of POIs (e.g. POI 1 and 2 in Figure 4 are covered for POI 1 in Figure 5). The same behavior was found in the remaining users. Then, in Figure 6 we present the average amount of POIs found per user when we considered different sliding window size and cluster radius. On average each user has 2.15 POIs; the minimum per day was 0, and the maximum was 6. This amount of POIs was found when the cluster radius was set to 500 m., and a 4 weeks sliding window was considered. In contrast, each user has 0.57 POIs when we considered a cluster radius of 100 m., and a 8 weeks sliding window. When the sliding window was set to 8 weeks, a lower number of POIs was found for the three values of the cluster radius. We can attribute the decrement of POIs due to the fact that considering the user mobility for a long time, just a few places are always visited by the user (e.g. home, work); in contrast, other places are just significant for a short time period.

Table 1: Location data statistics according to the day of the week (M represents Monday; T represents Tuesday, and so on)

	M	T	W	T	F	S	S
% of day in POIs	42.95	52.52	52.78	52.58	49.58	44.90	33.50
Avg # weeks	15	16.29	16.35	17.47	16.64	16.58	18
Max # weeks	82	94	84	87	89	87	74
Min # weeks	9	12	11	13	13	11	9

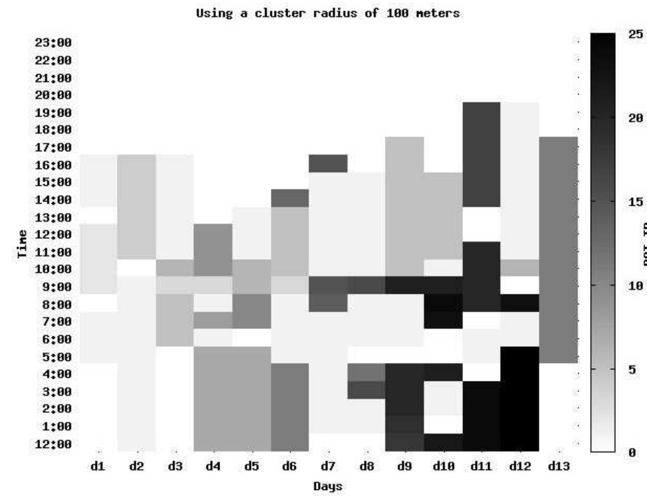

Figure 4: Mobility of user 2 over all Tuesdays

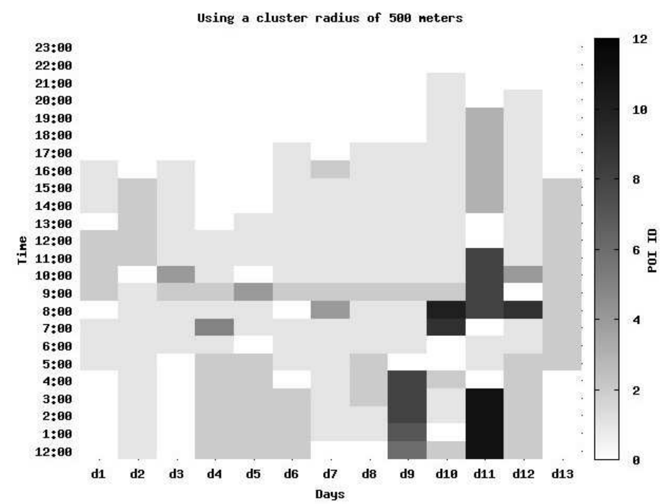

Figure 5: Mobility of user 2 over all Tuesdays.

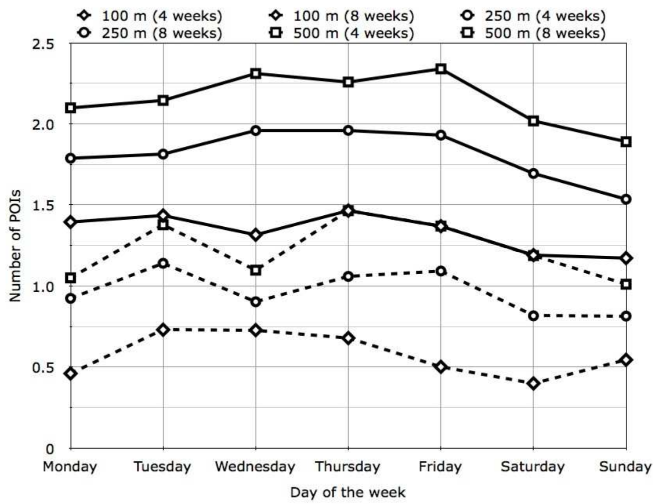

Figure 6: Average of POIs found for each day.

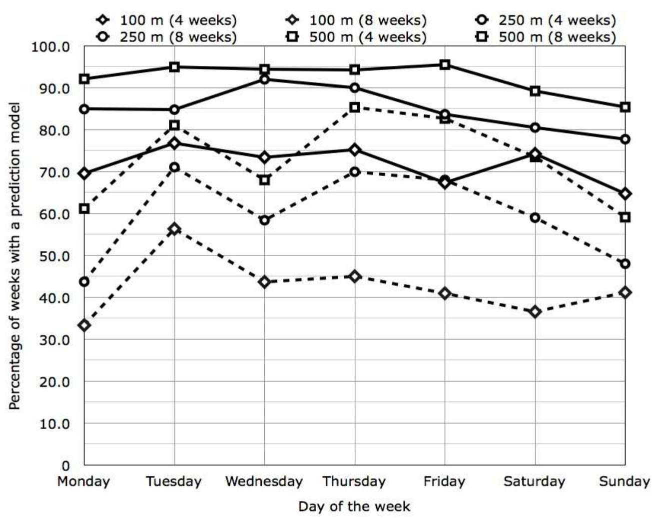

Figure 7: Percentage of weeks with a prediction model.

Considering a cluster of 500 m, a 4 weeks sliding window, and grouping POIs by weekdays and weekends, on average people have more POIs on weekdays (2.23); in contrast, people have 1.95 POIs on weekends. The highest number of POIs was found on Fridays (2.33), and the lower was found on Sundays (1.89). Even though for each user there is not a great amount of POIS, and so neither a great amount of GPS data, the users spend a lot of their time at their POIS. Table 1 presents the percentage of time that the user spends at POIS according to the day.

Predicting user location over time
Table 1 presents the average, minimum, and maximum values of weeks available [1] for testing each prediction model over time. Then in Figure 7 we present the percentage of these weeks having a prediction model. That is, weeks for which there were identified POIs within the sliding window and the prediction model was defined. As can be seen, the greatest accuracy was obtained when the sliding window was set to 4 weeks; in contrast, accuracy decreases when the sliding window was set to 8 weeks. For lack of space, in Figure 8 we just present the average accuracy obtained when the cluster was set to 500 m. and a sliding window of 4 weeks. This accuracy represents the average obtained after updating the prediction models over all weeks available for each user. When a prediction period (ΔT) of 1 hour was used, an accuracy of 41 % was obtained; 38 %, 39%, and 43 % for a ΔT of 2,4, 8 hours respectively. Also, we obtained results of up to 77 %, 72 %, 72 %, and 69 %, for the respective periods. As we can see in Figure 8, there is not a unique best value for the observations, for a ΔT of 1 and 2 hours, the best value was 2 hours; for a ΔT of 4 hours, the best value was 4 hours; finally, for a ΔT of 8 hours, the accuracy is very similar.

[1] Weeks refer to the amount of records available for each day

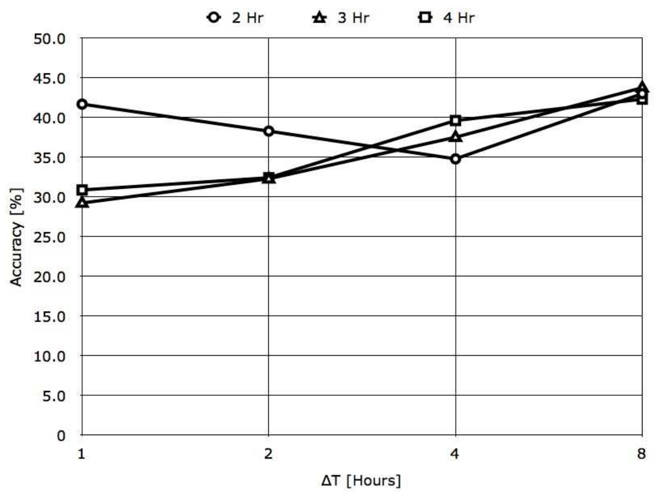

Figure 8: Results obtained using a sliding window of 4 weeks, and a cluster radius of 500 m.

After analyzing the results, we have some comments. The fact that the best accuracy was found considering a ΔT of 8 hours, can be explained because some observations are considered in that period. When considering a ΔT of 1,2 or 4 hours, these prediction periods could be included in just one observation. In this case, the predicted POI would be the one most likely in that prediction period. However, if a longer ΔT is used, more observations will be considered and the probabilities of the transition and confusion matrix will be important to define the sequence of POIs to be visited. Also, taking into account the above statement, it would be desirable to consider a longer ΔT to know the performance of the prediction model. However, most of GPS trajectories are distributed over a limited period of the day. Finally, although we do not present the results when the cluster radius was set to 250 and 100 m, there is a significant decrement on the accuracy, about 20 % and 30 % respectively (using a sliding window of 4 weeks). And, for a sliding window of 8 weeks, an accuracy of 30 % was obtained (averaging the accuracy for the 3 different cluster radius).

Conclusions and future work

In this paper we presented a spatio-temporal prediction model to forecast user location in a medium-term taking advantage of the Markovian property shown in user's mobility. The focus of our work is in updating the prediction model taking into account changes in user mobility. The technique we used is a sliding window with a threshold capturing a minimum amount of time where user mobility has a pseudo stationary behavior. Even though the selected dataset does not have a great amount of location data, and each prediction model was updated several times, this evaluation shows that our prediction model can yield good accuracy to predict the user mobility in a medium-term for up to 8 hours. The underlying empirical hypothesis are the relationship between three characteristics of mobility (week day, time of the day, and current location) and the use of a sliding window (the last 4 weeks) to train the prediction model. We recognize that this method cannot be applied for everyone; there are people with highly dynamic agendas (e.g. politicians), and the mobility patterns they exhibit are much more complicated. We obtained a low accuracy as we expected; however, it can be improved considering more contextual data. Instead of a sliding window, it would be appropriate to identify mobility patterns for specific time periods. For example, using data from scholar activities, social networks, and user profile, different prediction models can be defined for a student. Then, using these models, we can switch among them according to the current date, or take them as a baseline to make little changes to capture current user behavior. These aspects are being considered in an ongoing work.

References

[1] Alvarez-Lozano, J., García-Macías, J. A., and Chávez, E. User location forecasting at points of interest. In *Proc. RecSys* (2012), 7–12.

[2] Ashbrook, D., and Starner, T. Using GPS to learn significant locations and predict movement across multiple users. *Personal Ubiquitous Computing 7*, 5 (Oct. 2003), 275–286.

[3] Eagle, N., and Pentland, A. Reality mining: sensing complex social systems. *Personal Ubiquitous Computing 10*, 4 (March 2006), 255–268.

[4] Gonzalez, M. C., Hidalgo, C. A., and Barabasi, A.-L. Understanding individual human mobility patterns. *Nature 453*, 7196 (June 2008), 779–782.

[5] Kang, J. H., Welbourne, W., Stewart, B., and Borriello, G. Extracting places from traces of locations. *SIGMOBILE Mobile Computing and Communications Review 9*, 3 (July 2005), 58–68.

[6] Kim, M., Kotz, D., and Kim, S. Extracting a mobility model from real user traces. In *Proc. INFOCOM* (2006), 1 –13.

[7] Krumm, J., and Horvitz, E. Predestination: inferring destinations from partial trajectories. In *Proc. UbiComp*, Springer-Verlag (2006), 243–260.

[8] Markov, A. A. *Theory of Algorithms*. Israel Program for Scientific Translations, Bloomington, USA, 1961.

[9] Mathew, W., Raposo, R., and Martins, B. Predicting future locations with hidden markov models. In *Proc. UbiComp*, ACM (2012), 911–918.

[10] Monreale, A., Pinelli, F., Trasarti, R., and Giannotti, F. Wherenext: a location predictor on trajectory pattern mining. In *Proc. SIGKDD*, ACM (2009), 637–646.

[11] Motahari, S., Zang, H., and Reuther, P. The impact of temporal factors on mobility patterns. In *Proc. HICSS*, IEEE (2012), 5659–5668.

[12] Rabiner, L. R. A tutorial on hidden Markov models and selected applications in speech recognition. *Proc. of the IEEE 77*, 2 (May 1989), 257–286.

[13] Ram, A., Jalal, S., Jalal, A. S., and Kumar, M. A density based algorithm for discovering density varied clusters in large spatial databases. *Int. J. of Computer Applications 3*, 6 (June 2010), 1–4. Published By Foundation of Computer Science.

[14] Sadilek, A., and Krumm, J. Far out: Predicting long-term human mobility. In *Proc. AAAI*, AAAI Press (2012).

[15] Scellato, S., Musolesi, M., Mascolo, C., Latora, V., and Campbell, A. T. NextPlace: a spatio-temporal prediction framework for pervasive systems. In *Proc. Pervasive*, Springer-Verlag (2011), 152–169.

[16] Viterbi, A. J. A personal history of the Viterbi algorithm. *IEEE Signal Processing Magazine 23*, 4 (July 2006), 120–142.

[17] Vu, L., Do, Q., and Nahrstedt, K. Jyotish: A novel framework for constructing predictive model of people movement from joint wifi/bluetooth trace. In *Proc. PERCOM*, IEEE (2011), 54–62.

[18] Yavas, G., Katsaros, D., Ulusoy, O., and Manolopoulos, Y. A data mining approach for location prediction in mobile environments. *Data & Knowl. Eng. 54*, 2005 (August 2004), 121–146.

[19] Zheng, Y., Li, Q., Chen, Y., Xie, X., and Ma, W.-Y. Understanding mobility based on GPS data. In *Proc. UbiComp*, ACM (2008), 312–321.

[20] Zheng, Y., Xie, X., and Ma, W.-Y. Geolife: A collaborative social networking service among user, location and trajectory. *IEEE Data Engineering Bulletin 33*, 2 (2010), 32–39.