ARTICLE IN PRESS

Pervasive and Mobile Computing ▮ (▮▮▮▮) ▮▮▮–▮▮▮

Contents lists available at ScienceDirect

Pervasive and Mobile Computing

journal homepage: www.elsevier.com/locate/pmc

A probabilistic kernel method for human mobility prediction with smartphones

Trinh Minh Tri Do [a,*], Olivier Dousse [b], Markus Miettinen [b], Daniel Gatica-Perez [a,c]

[a] Idiap Research Institute, Martigny, Switzerland

[b] Nokia Research Center, Lausanne, Switzerland

[c] EPFL, Lausanne, Switzerland

ARTICLE INFO

Article history:
Received 13 January 2014
Received in revised form 27 August 2014
Accepted 6 September 2014
Available online xxxx

Keywords:
Smartphone
Human mobility
Prediction
Personalized service

ABSTRACT

Human mobility prediction is an important problem that has a large number of applications, especially in context-aware services. This paper presents a study on location prediction using smartphone data, in which we address modeling and application aspects. Building personalized location prediction models from smartphone data remains a technical challenge due to data sparsity, which comes from the complexity of human behavior and the typically limited amount of data available for individual users. To address this problem, we propose an approach based on kernel density estimation, a popular smoothing technique for sparse data. Our approach contributes to existing work in two ways. First, our proposed model can estimate the probability that a user will be at a given location at *a specific time in the future*, by using both spatial and temporal information via multiple kernel functions. Second, we also show how our probabilistic framework extends to a more practical task of location prediction for *a time window in the future*. Our approach is validated on an everyday life location dataset consisting of 133 smartphone users. Our method reaches an accuracy of 84% for the next hour, and an accuracy of 77% for the next three hours.

1. Introduction

The advances in mobile sensing and computing have enabled the integration of machine learning into personal mobile devices. In particular, smartphones emerge as all-purpose devices with personalized services, where the personalization is based on what the smartphone knows about the user. Smartphones can unobtrusively collect data about where users go and what they do, and build a detailed understanding of the user. First, the recorded data can be used to characterize multiple aspects of the user including demographic information [1,2] or personality [3]. Second, clustering methods can be applied to extract recurrent user contexts such as commonly visited places [4], providing a high level representation of context (instead of raw measurements). Finally, along with extracting and organizing information from the past, the phone can also learn a behavior model that can predict future activities and venues.

Location prediction can benefit mobile applications and services by letting the applications adapt to possible movements of the user. This can help a mobile device, for instance, to adapt its user interface based on the anticipated locations that the user will visit during the course of a day. As one example, it can prefetch and display relevant information related to the

* Corresponding author. Tel.: +41 27 721 77 51.
E-mail addresses: tri.do@idiap.ch, do@idiap.ch (T.M.T. Do), olivier.dousse@here.com (O. Dousse), markus.miettinen@trust.cased.de (M. Miettinen), gatica@idiap.ch (D. Gatica-Perez).

http://dx.doi.org/10.1016/j.pmcj.2014.09.001
1574-1192/© 2014 Published by Elsevier B.V.

predicted target locations. Note that, for such scenarios, personalization is key as the interest does not lie in predicting the places that people are likely to visit, but rather in anticipating the movements of a *single user*. Also, since location traces of users are highly privacy-sensitive, it is not desirable to rely on a solution that requires location traces to be aggregated to a central data storage. Therefore, the prediction method has to be such that it relies only on the context history of a specific user for whom the prediction will be made.

Previous studies on mobility prediction have usually focused on predicting the next place where a user goes [5]. However, in practice, the prediction capability needs to go beyond the anticipated next place of the user, and instead provide predictions for different look-ahead periods. This is because applications might want to provide different kinds of information, depending on how imminent the user's visit to a location will be. For example, an application may want to prefetch traffic information for the route to a place that the user is predicted to visit during the next few hours, but for visits farther away in the future (during the next 24 h) other information like a weather forecast for the target location might be more appropriate. With these requirements in mind, we develop a flexible prediction method which can predict user location for a given timestamp or for different look-ahead time windows.

The mobility prediction problem can be formalized as a contextual prediction problem where the future movements are assumed to depend only on the user context, which is characterized by space and time in this paper. The assumption is based on the repetitive nature of human mobility: similar contexts might imply similar movements in the future. For example, from the mobility traces of a given user, one might observe that if he is at a given train station around 8:00 AM on Monday then he will likely be at work around 8:25 AM. Under a probabilistic framework, the location prediction task consists on estimating the conditional distribution over the set of future location candidates for a specific context, based on mobility history. This can be modeled by representing the user context as a combination of discrete states (e.g., at place X at hour Y on day Z), and so the conditional distribution is proportional to the counts of possible outputs for the considered context. This approach, however, suffers from a major issue with discretization: the relationship between states are lost. For example, if we discretize the time of day into 24 time slots by hour, then 7:59 AM and 8:00 AM belong to two completely different time slots, while they are actually very close. We resolve this problem by using kernel density estimation (KDE), a non-parametric approach for the estimation of the conditional probabilities. The idea is to use kernel functions to measure the similarity between the current context and data points in the location history. Data points with the highest similarity scores will have significant impact on the outputs. This approach is advantageous for dealing with sparse data, which happens when the amount of data is limited or when the user is in an infrequent context.

Our paper makes two contributions. First, we propose a non-parametric approach for location prediction based on kernel density estimation, for which we introduced several kernels to capture different aspects of spatio-temporal context. Our probabilistic framework can make predictions for a specific time or for a look-ahead time-window without any heuristics. Second, we present a thorough application-oriented study of location prediction, which considers the look-ahead time interval as a key aspect. Our analysis is conducted on a real life dataset with state-of-the-art spatial resolution and longitudinal recording period. Our experimental results show how the prediction performance is affected by various factors such as the time of the day or the look-ahead time window.

The paper is organized as follows. The next section discusses related work on mobility prediction in the context of mobile computing and compares our contributions with respect to the existing literature. Section 3 presents our prediction frameworks, with formal descriptions of the data representation and the prediction task, that is to predict user location at a given time in the future. Our analysis starts in Section 4, which introduces the location dataset. We report baseline results in Section 5. The results reveal the contexts for which the baseline performance is low, and motivates our proposed KDE approach presented in Section 6, a probabilistic model which uses spatio-temporal context. While the proposed method improves the accuracy on difficult settings such as large look-ahead time, we also found situations in which a simple baseline works best. Section 7 thus presents our final solution, which is a combination of the proposed model and a probabilistic version of the baseline method. In Section 8, we generalize the framework from predictions for a specific time, to predictions for a time window, reporting experimental results with information retrieval measures appropriate for this new task. Finally, Section 9 provides concluding remarks.

2. Related work

Human mobility analysis has become an active research topic thanks to the development of location tracking techniques [6,7]. Song et al. [8] presented a study on predictability of human mobility by analyzing the entropy of location traces. The analysis of entropy shows that the limit of predictability is around 93% for hourly sequences of GSM cell IDs, where the average size of a cell's area is about 3 km^2. Jensen et al. [9] applied the same methodology for analyzing predictability of discrete time series coming from several sources including GSM, WLAN, Bluetooth, and accelerometer. Recently, Lin et al. [10] extended the original work by studying the effect of spatio-temporal scales on predictability, showing that predictability increases with spatial scale and decreases with temporal scale.

Several prediction methods have been proposed for human mobility in different contexts (i.e., using different devices and sensors) and with different definitions of the prediction task. Some notable works are listed in Table 1. In transportation, Krumm et al. [12] consider the problem of inferring the destination based on partial paths which could be applied in navigation assistance systems. For example, context-aware trip recommendations can be produced by combining user specific needs (e.g., finding a gas station) with the inferred primary destination [19]. At a higher level, the prediction task

Table 1
Related works on human mobility prediction.

Study	Prediction task	Data type	Data collection devices	Population and duration
Markov models [5]	next location	GPS	GPS device	7 users, 3–7 months
String based prediction [11]	next location	WiFi	543 WiFi APs within a campus	6000 users, 2 years
Predestination [12]	trip destination	GPS	GPS-equipped car	169 subjects, two weeks
Present/absent probability [13]	home/away prediction	GPS	smartphones	34 users, 8 weeks
NextPlace [14]	future location/stay duration	Multiple datasets: (1) GPS data from 252 taxis, 23 days; (2) GPS data from 19 smartphone users, 12 days; (3) WiFi data collected with WiFi APs, 2043 users, 60 days; (4) WiFi data collected with WiFi APs, 804 users, 370 days		
Contextual conditional model [15]	next location/stay duration	GPS, Bluetooth, WiFi, Call logs	smartphone	153 users, 17 months
Mobile Data Challenge [1,16–18]	next location	rich smartphone data	smartphone	80 users, 18 months
This work	future location	GPS, WiFi	smartphone	133 users, 3–18 months

is to infer people's movement among places such as "*If the user is currently at home, which are places that he will visit today*?". However, there are differences on how exactly the prediction task is defined. In an early analysis with GPS traces of seven users over several months [5], Ashbrook et al. proposed to extract significant places and represent location traces as strings, then use Markov models to predict the next place that a user will visit. Song et al. [11] investigated various prediction methods on symbolic location traces collected with WiFi access points of a university campus. A few works have attempted to improve the prediction performance by exploiting other smartphone data besides location, such as call logs, Bluetooth, and application usage [15]. However, it is still challenging to efficiently exploit these additional information for location prediction. In the next place prediction task of the Mobile Data Challenge 2012 [1], the best methods relied only on spatio-temporal information to predict future location [16–18]. Closely related to our work, Scellato et al. [14] address the problem of predicting user location at a given time in the near future (e.g., in several hours) instead of the next movement as studied in [1,15]. However, our work differs from [14] in the following aspects: (1) in addition to learning the time distribution for each place, we also exploit the transition patterns between places by using spatial kernels. (2) We investigate the prediction problem on more complete and dense data. For example, our data has five times more places per user than the CenceMe dataset [20] and the percentage of staying time in extracted places is much higher (67% vs. 15%). (3) Instead of using time series to infer next visiting times of each place, then combining predictions with a heuristic method, we propose a probabilistic method which estimates directly the conditional probability of a location at a given time. These probabilities provide generic scores for high-level tasks such as the information retrieval task of predicting the most likely set of places for a time interval.

3. Prediction framework

Our framework is built on a high-level representation where location traces are encoded as a history of place visits, for which we only keep visits of significant time periods (e.g., a few minutes) to filter out places that people pass by but did not actually visit. Ideally, one would define places as physical addresses or rooms, which match perfectly the definition of the place in real life. However, this would require an accurate positioning capability for both indoor and outdoor that is beyond what current location tracking systems can provide. For smartphone-based location systems using GPS/WiFi (which are used in this work), the place extraction can output locations that correspond to regions of about 100-meter radius.

Table 2 illustrates how location traces are stored in our data. Formally, the history is stored as a sequence $H = (t_i, l_i)_{i=1...n}$, where t_i is a timestamp that indicates when the user arrived at place l_i, l_1 is the first visited location in the history, and l_n is the most recent location of the user (thus it is also the current location). The sequence $\{t_i\}$ must be in increasing order and by construction $l_{i-1} \neq l_i$ for all i. Furthermore, l_i is a positive integer corresponding to a place ID, but it can be negative in some special cases:

- *TRANS* $= -1$: the location is unknown or not a significant place. This happens when the user is on the move (i.e., he briefly passes by many non-significant places in the trajectory).
- *OFF* $= -2$: the phone or the sensing module is off.

Prediction problem formulation. We are interested in predicting user location at a specific time in the near future. This can be formalized as follows: *At time t, we want to predict user location at time $t + \Delta t$, given his history of place visits up to time t, denoted by $\mathcal{H}_t$*. The history of visits $\mathcal{H}_t$ can be viewed as a training dataset from which we can extract repetitive mobility patterns such as the fact that the user arrives at work around the same time every working day. The time interval Δt is the look-ahead time of the prediction, which varies from 5 min to 24 h in our analysis. Intuitively, predicting the near future (for example, the next hour) is easier than predicting user location a long time ahead (for example, in 3 h). Fig. 1 illustrates the prediction setting, where blocks represent place visits and the horizontal axis corresponds to time.

Recall that locations are encoded by abstract place IDs instead of geo-location, so that the set of possible outputs of the prediction algorithm is the set of place IDs that the user had visited up to the prediction time (including the *TRANS* code). Note that the ground truth location may not belong to the considered output set when people visit new places, but we do

Please cite this article in press as: T.M.T. Do, et al., A probabilistic kernel method for human mobility prediction with smartphones, Pervasive and Mobile Computing (2014), http://dx.doi.org/10.1016/j.pmcj.2014.09.001

Table 2
Example of location history.

Timestamp (t_i)	PlaceID (l_i)	
01-01-2012 12:15:03	246	(a restaurant)
01-01-2012 13:35:02	−1	(transition)
01-01-2012 14:12:15	204	(a friend's place)
01-01-2012 17:13:15	−1	
01-01-2012 17:40:13	18	(home)
01-01-2012 20:11:10	−2	(phone off)
01-01-2012 22:17:15	18	
02-01-2012 08:15:12	−1	
02-01-2012 08:30:12	376	(office)
02-01-2012 12:30:16	275	(another restaurant)
02-01-2012 13:25:56	376	
...	...	

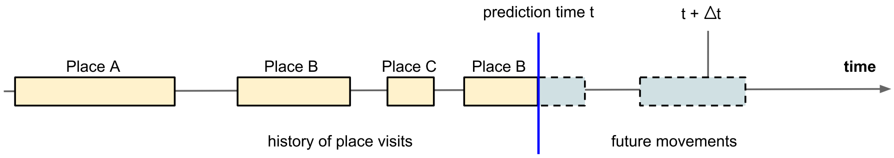


Fig. 1. Real-time prediction scenario in which the prediction time t is continuous and the history of place visits increases over time. The fundamental task is to predict the location at time $t + \Delta t$.

not introduce a special code "new place" since it corresponds to a small fraction of the data. Based on our collected location traces, we estimate that the average probability of being in a new place at a random time within the next 24 h is $p = 0.03$.

As a simulation of a real-life scenario in which the application makes predictions on the fly, we consider the prediction time t to be continuous. For a single user data, there are multiple training sets which correspond to different values of t. Since the training set size increases with t, we could expect that the predictability of user movements improves over time.

4. Location data

Our experiments were performed on the data from the Lausanne Data Collection Campaign (LDCC) which was ran from October 2009 to the end of March 2011 in Switzerland [21]. About 180 volunteer users around Lake Léman participated in the campaign. LDCC participants were asked to carry Nokia N95 phones with recording software running in the background. Thanks to a dynamic sampling technique using a state machine approach, the phone recorded data continuously on a 24/7 basis with the only restriction of having to charge the phone once a day.

Location sensing. The raw location traces were collected by combining GPS and WiFi readings. Since GPS is a power-hungry sensor, the recording software only activates GPS (one reading per 10 s) when the phone is detected to be moving. As a complement to GPS data, WiFi readings are made to track user location indoors. The recording software estimates the position of each observed WiFi Access Point (AP) based on GPS readings that are close to AP reading (time difference of less than 90 s).

Extracting the history of place visits. The raw location data points were transformed into a high level representation based on a two-step process proposed by Zheng et al. [22]. The details of our implementation can be found in our earlier work [23]. In this work, a location trace is first segmented into transitions and stay points, the minimum time of stay points was set to 10 min. Then the set of stay points are clustered into stay regions of 100-meters radius using a grid clustering algorithm. The set of extracted stay regions are used to define places that the user visited. Places are extracted for each user independently of the data of other users.

Data filtering and statistics. To investigate the prediction task, we filtered out outliers, these were users whose recorded location traces were very incomplete due to technical issues. This filtering step is necessary to avoid biased estimates of prediction performance. At the end, there were $N = 133$ users having location traces satisfying three constraints: (a) 30% of their days with location data; (b) recording time of at least 90 days; (c) a fraction of missing location data and transitions of less than 70%. Samples of the data are illustrated in Fig. 2. The recording time of location traces are plotted in Fig. 3, which shows a correlation between the recording time and the number of places that people had visited. On average, each user had visited 75.8 places during the recording period. People were detected to stay 67% of the time, moving 5% of the time, while the amount of missing data were 28%.

5. How difficult is the prediction task? Baseline performance

In this section, we establish baseline performance for our prediction task based on two basic observations from the mobility data. First, while people usually visit a large number of places in everyday life, their location traces are dominated by

Please cite this article in press as: T.M.T. Do, et al., A probabilistic kernel method for human mobility prediction with smartphones, Pervasive and Mobile Computing (2014), http://dx.doi.org/10.1016/j.pmcj.2014.09.001

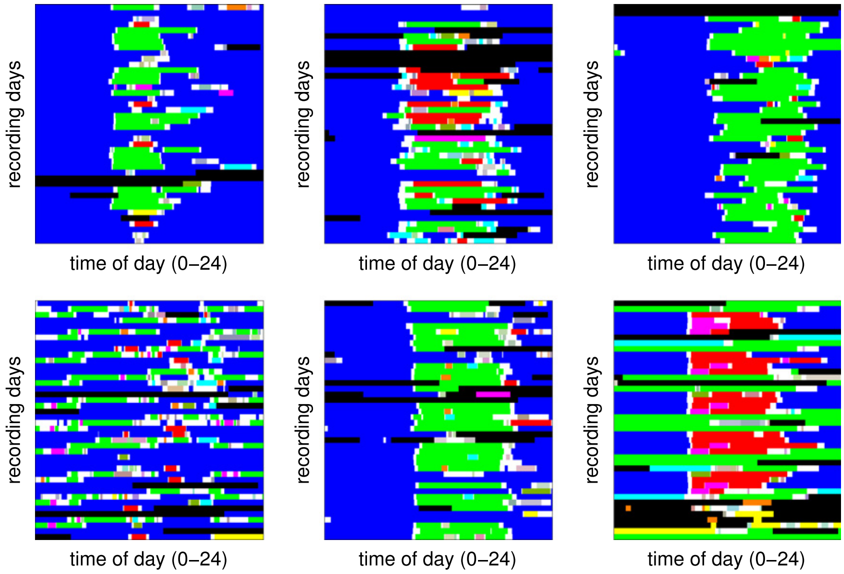


Fig. 2. Sample of location data from 6 different users. Rows correspond to days. Black represents missing data, white represents transition state, and other colors correspond to different places that the user visited. Note that places are user-dependent, and that blue corresponds to the user's home. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

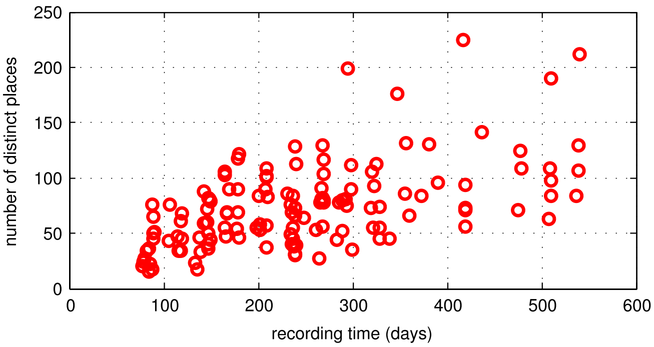


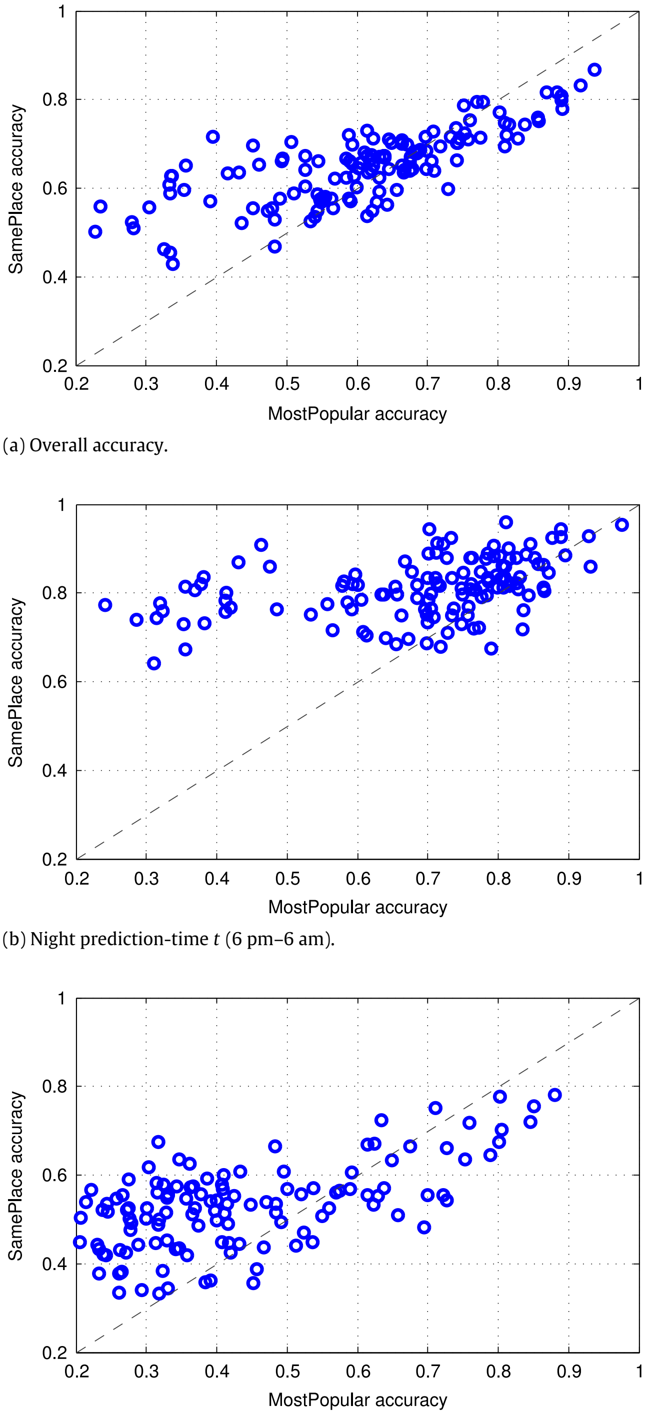
Fig. 3. Scatter plot of recording time and number of distinct places, each circle represents a single user.

a small number of significant places [23]. This observation leads to our first baseline model called *MostPopular* which predicts future location based on the popularity of visited places. The second observation is that people tend to stay some time in each place, instead of continuously jumping from one place to another. Therefore, a method predicting that the user will be at the same place, called *SamePlace*, for the near future will have a high accuracy. These two methods are described below:

MostPopular method: This method predicts the future location at time $t + \Delta t$ to be the most popular place in the observed history regardless of the context. The most popular place is defined as the place with the largest stay time up to the prediction time t. Note that the most popular place can vary over time t (especially at the beginning of the location trace), but it usually converges to the home location after a few days of data collection.

SamePlace method: This method predicts the future location at time $t + \Delta t$ to be the same place as the location at time t regardless of Δt. Note that if the current location is not available (due to missing data) then this method is unable to predict the future location.

The evaluation was done as follows. First, we generated a set of prediction times t for each location trace, one timestamp every 5 min. Then for each Δt of interest, we predict the future location at time $t + \Delta t$ with the above methods. Finally, the accuracy of a method is the fraction of correct predictions over the total number of predictions. Fig. 4 shows the baseline accuracies of predicting user location in the next 3 h (i.e., $\Delta t = 3$ h) for each user. Looking at the overall accuracies in Fig. 4(a), we see that the two baseline methods are competitive, reaching an accuracy of about 0.65. While these baseline results are relatively high, they are biased by the night periods in which people generally sleep at home so that the prediction is accurate. The effect of time on prediction is clearly highlighted in Fig. 4(b), (c) which illustrates the prediction accuracy for daytime (6 am–6 pm) and night time (6 pm–6 am) separately. As expected, the prediction accuracy for night time is very high (around 0.81) even with very simple methods. For daytime, the baseline accuracy drops to 0.52, with SamePlace



(a) Overall accuracy.

(b) Night prediction-time t (6 pm–6 am).

(c) Day prediction-time t (6 am–6 pm).

Fig. 4. Accuracies of predicting location in the next 3 h with two baseline methods ($\Delta t = 3$ h). Each data point corresponds to a user.

method being more accurate than MostPopular method. While the prediction task is much more challenging for daytime, this is the period of the day in which users are more active and predictions are probably more useful.

Up to now, we only consider a fixed look-ahead time (Δt) of 3-hours. While the performance of the MostPopular method is not affected by Δt, the performance of SamePlace method depends significantly on this variable. Fig. 5 reports prediction accuracy as a function of Δt in log-scale, showing that the probability of being at the same place is very high for small Δt, and decreases as Δt increases from minutes to hours. However, the curve has peaks at multiples of 24-hours, reflecting the

Please cite this article in press as: T.M.T. Do, et al., A probabilistic kernel method for human mobility prediction with smartphones, Pervasive and Mobile Computing (2014), http://dx.doi.org/10.1016/j.pmcj.2014.09.001

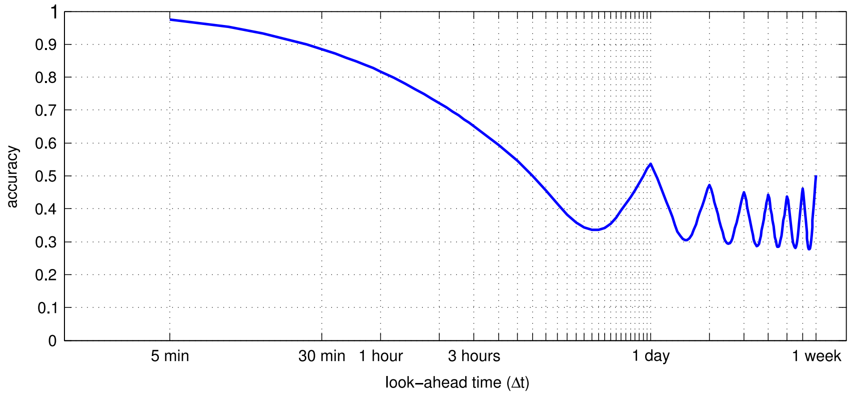


Fig. 5. Accuracy of the SamePlace baseline on the LDCC data.

fact that there is a high chance that the user will be found at the same place at the same time of the following day, the day after, etc. The accuracy of the SamePlace model for predicting location in the next 24-hours is 0.54. This observation suggests that the combination of spatial context (current place) and temporal context (current time) is relevant for predicting future locations, even for large Δt. In the next section, we will show how to exploit spatial and temporal context in a prediction model based on a combination of a naive Bayes assumption and a kernel density estimation method (KDE).

6. Our approach: spatio-temporal probabilistic model

There are several factors that explain why a person is located at a specific place and a given time, including regular routines (such as home-work), needs (such as lunch time), and social relations (such as hanging out with friends). While these factors are highly personal and cannot be entirely integrated in an automatic sensing framework, some of them can be captured using smartphone sensors and statistical methods. We focus on temporal and spatial information and learn the dependencies between these contextual variables and the future location of the user. Temporal context captures regular mobility patterns from the weekly calendar, such as being at a particular place at a given time of the day and day of the week. Furthermore, the spatial information is helpful when we aim at exploiting the dynamics of movement, such as going to one specific place after visiting a given place. In the remainder of this section, we first describe how to incorporate different contextual variables in a single probabilistic model. Then we show how hyper parameters are determined using a calibration dataset. Finally, we evaluate the spatio-temporal model and compare it to the baseline methods.

6.1. Probabilistic framework

Let $\mathbf{c}$ be all contextual information for the prediction of location at time $t + \Delta t$. Note that the context $\mathbf{c}$ depends on the prediction time t and the look-ahead time Δt, but we drop the time components to simplify the presentation. To ease the model, we assume that the context information is represented as a fixed length vector where c_f denotes the fth element. To compute the conditional probability of a location l given the context $\mathbf{c}$, we use Bayes' theorem:

$$P(l|\mathbf{c}) = \frac{P(l)P(\mathbf{c}|l)}{P(\mathbf{c})} \propto P(l) \prod_{f} P(c_f|l) \tag{1}$$

in which the elements of $\mathbf{c}$ are assumed to be independent given the future location l. By this formulation, the conditional probability can be factorized into $P(c_f|l)$, which are probabilities that a specific context occurs when a user is in place l. Compared to the original conditional probability, the elementary probability is "easier" to estimate as it involves fewer random variables. Our estimation method which is based on KDE is described below.

The probability $P(c_f|l)$ can be estimated from the mobility history by sampling data from visits of place l. Note that c_f can be discrete (e.g., the day of the week) or continuous (e.g., the time of the day) in our framework. One can use a parametric approach, which assumes that c_f random variable follows a certain distribution (such as a Gaussian mixture) whose parameters need to be estimated from the data. As the sampled dataset evolves from "sparse" to "dense" (if the place l is frequently visited), this approach requires that we adapt the hypothesis distribution to the density of the data: we should not learn a complex distribution from only a few samples nor to use an oversimplified distribution when the data is dense. Kernel density estimation is advantageous in this context, as this non-parametric approach does not require any hypothesis distribution. Intuitively, for a given independent and identically distributed sample, KDE estimates the unknown probability density function by placing a little bump on each training point and summing them. The bump is defined by a smoothing kernel, which takes the "distance" of two data points as input. In this way, the estimated density is high at the area which is close to many data points, and the density function can have any shape depending on the data and the kernel. Note that

Please cite this article in press as: T.M.T. Do, et al., A probabilistic kernel method for human mobility prediction with smartphones, Pervasive and Mobile Computing (2014), http://dx.doi.org/10.1016/j.pmcj.2014.09.001

Table 3
List of kernels for various contextual information types. The $u \ominus v$ notation denotes the time difference between two times of the day u and v. For temporal information, the time unit of variables u and v is days.

Kernel code	contextual information c_f	Kernel type	Parameters
TOD	time-of-day $c_f \in (00:00, 23:59)$	$K(u, v) = \mathcal{N}(u \ominus v; 0, \sigma^2)$	$\sigma \in R^+$
DOW	day-of-week $c_f \in \{1(Mon), \dots, 7(Sun)\}$	$K(u, u) = 1 - \lambda, K(u, v) = \frac{\lambda}{6}$ $\forall v \neq u$	$\lambda \in (0, 1)$
WE	is-weekend $c_f \in \{0(weekday), 1(weekend)\}$	$K(u, u) = 1 - \lambda,\ K(u, v) = \lambda$ $\forall v \neq u$	$\lambda \in (0, 1)$
TimeD	time-of-day $c_f \in (00:00, 23:59)$	$K(u, v) = \begin{cases} 1 - \lambda & \text{if } u \ominus v \leq \tau \\ \lambda & \text{if } u \ominus v > \tau \end{cases}$	$\lambda \in (0, 1)$ $\tau \in (0, 1)$
Place	place IDs at time: $(t - 5m, t - 10m, ..)$ $c_f = (c_{f\,1}, \dots, c_{f\,L}) \in \{placeIDs\}^L$	$K(u, v) = \frac{1}{Z} \prod_j \frac{1}{2^j} K'(u_j, v_j)$ $K'(u_j, v_j) \begin{cases} 1 - \lambda & \text{if } u_j = v_j \\ \lambda & \text{if } u_j \neq v_j \end{cases}$	$\lambda \in (0, 1)$

for discrete variable, we can use the Kronecker delta as a distance measure instead of the Euclidean distance which is used for continuous variables.

As discussed above, the training data is generated from the history of place visits. In practice, we use uniform sampling in which data points are generated every five minutes. The history of visits is then transformed to a pair of a contextual matrix **C** and a corresponding location vector **L**,

$$\mathbf{C} = \begin{bmatrix} \bar{c}_{11} & .. & \bar{c}_{1F} \\ .. & .. & .. \\ \bar{c}_{m1} & .. & \bar{c}_{mF} \end{bmatrix}; \qquad \mathbf{L} = \begin{bmatrix} \bar{l}_1 \\ .. \\ \bar{l}_m \end{bmatrix}, \tag{2}$$

where m is the number of generated data points (e.g., one data point every 5 min), F is the number of extracted contextual features, $\bar{c}_{if}$ denotes the fth element of the contextual vector computed for the ith data point, and $\bar{l}_i$ denotes the user location corresponding to ith data point. The probability $P(c_f|l)$ can be estimated as follows:

$$P(c_f|l) = \frac{1}{J(l)} \sum_{i \in J(l)} K_f(c_f, \bar{c}_{if}), \tag{3}$$

where $J(l) = \{i|\bar{l}_i = l\}$ is the set of indices of data points generated from visits to place l, and $K_f(.,.)$ is a kernel defined on pairs of fth contextual variables that will be detailed in the next subsections. The size of the **C** and **L** matrices increases linearly over time, meaning that the storage and computational costs increase linearly as well. In our experiments, we keep these matrices increasing continuously, but in practice, we can limit their size by introducing an expiration date for each data point.

KDE has been used in a previous study for modeling the waiting time distribution for a given place to be revisited [24], in which a Gaussian kernel is applied on the temporal variable. Our use of KDE for location prediction is novel in the sense that we define a global conditional model for all places instead of considering one model for each place. Furthermore, our kernel method is not restricted to one random variable, the factorized probabilistic model in Eq. (1) allows the combination of multiple temporal and spatial variables (if available, other contextual variables too).

6.2. Temporal and spatial kernels

We investigated a number of temporal and spatial kernels. A large number of kernels adds computational cost to the system, but it will also add more flexibility to the model for capturing effectively different mobility patterns. The list of kernels used in our work is summarized in Table 3, in which we provide the associated contextual information, the formulation of the kernel and the set of kernel parameters. For temporal context, we extract time-of-day, day-of-week, and weekend/weekday indicators. Note that we introduced several kernels to efficiently capture multiple aspects of the temporal context. As can be seen later in the experimental results, all temporal kernels were useful to some degree, especially for large Δt. The spatial context is represented as a fixed-length location sequence, sampled at different timestamps in the past. In our implementation, we use $L = 8$ timestamps ranging from 5 min to 24 h to summarize user movement during the past 24 h. More specifically, the spatial context at time t is the sequence of user locations at $t - 5$ min, $t - 15$ min, $t - 30$ min, $t - 1$ h, $t - 2$ h, $t - 4$ h, $t - 8$ h, $t - 24$ h, represented by a sequence of 8 place IDs. Note that user location might be unavailable for some timestamps, which is represented by the special code *OFF* discussed in Section 3.

Please cite this article in press as: T.M.T. Do, et al., A probabilistic kernel method for human mobility prediction with smartphones, Pervasive and Mobile Computing (2014), http://dx.doi.org/10.1016/j.pmcj.2014.09.001

We used four temporal kernels and one spatial kernel:

- TOD: a continuous kernel between times of the day, which is defined as a normal distribution over time-differences between the two timestamps with zero mean and variance σ^2.
- DOW: a discrete kernel between days of the week, parameterized by λ between 0 and 1.
- WE: a discrete kernel for day categories (weekend vs. weekday), parameterized by λ between 0 and 1.
- TimeD: a discrete kernel for the time difference between two timestamps u and v, parameterized by τ and λ. The kernel outputs two possible values depending on whether the time difference exceeds the threshold τ.
- Place: a discrete kernel between spatial contexts. The kernel between two location sequences can be factorized as a product of elementary kernels, computed for each timestamp of the two sequences. The elementary kernel between places is a discrete kernel parameterized by a single parameter λ. Note that the normalization constant Z can be omitted in the computation of $P(l|\mathbf{c}(t, \Delta t))$.

Besides the spatial kernel listed above, we also explored more sophisticated kernels based on the distance between places, note however that this kernel cannot be applied to abstract location such as WiFi fingerprint. A straightforward solution is to define a normal kernel over the geo-distances among places. We also developed a finer distance kernel by extracting a feature vector of each place (e.g., average staying time, visit frequency), and then using a multivariate normal distribution with diagonal covariance matrix to define the kernel between places. Unfortunately, these spatial kernels were not better than the discrete place kernel above and did not help to improve the performance of the whole system. For this reason, we did not include the results with these spatial kernels in the paper.

6.3. Kernel bandwidth optimization

Kernel bandwidths are hyper-parameters that define the smoothness of the estimated density functions. The larger the bandwidth, the smoother the density curve.

The set of parameters in Table 3 can be optimized automatically based on a training dataset. Each parameter is optimized sequentially to maximize the conditional likelihood on the training set using a heuristic search. Starting with an empty set of kernels (i.e., only use the prior), we iteratively add one kernel type and find the best parameters for that kernel on the sample data of a few users. Among an exponential number of options for kernel ordering, we chose to start with temporal kernels and end with the spatial kernel as in Table 3. At the first iteration, we optimize the parameter σ of the kernel *TOD* in the model with only 1 kernel. At the second iteration, the model has two kernels, *TOD* and *DOW*, while the σ of *TOD* is fixed, we optimize the parameter λ of *DOW*. The process iterates until the parameter λ of the last kernel (Place) is optimized. The order of kernel parameters to be optimized can, theoretically, affect the system since the optimization problem is not convex. Among an exponential number of possible kernel orders, we did some tests and did not find any significant changes in the results.

We also divide the space of look-ahead time Δt into multiple intervals, and then optimize the kernel parameter for each interval separately. The intuition behind this technique is that the importance of each kernel may vary depending on Δt. For example, spatial information can be important for predicting the next hour, but it is not very helpful for predicting location in 10 h. We implement this technique with 3 time intervals for Δt: from 0 to 1 h, from 1 to 3 h, and more than 3 h.

6.4. Evaluation of spatio-temporal model

In this section, we first study the contribution of each kernel to the overall accuracy of our model, and then we compare the method with the baseline results. To make the results generalizable, we always perform cross testing. In other words, to evaluate the prediction performance on a given user, we always use the set of parameters optimized on a training set that does not include that user. In general, the training data contains only 3 or 4 users, which is enough to find a good set of parameters. A few experiments with larger number of users show that the performance is not improved significantly.

Starting with a model with the TOD kernel only, we sequentially add more kernels to the model and study how the accuracy is improved. Fig. 6 shows the performance of our method with an increasing number of kernels. As can be seen, the combination of the two kernels TOD and DOW results in a competitive performance for predicting the next 3 h compared to the model that uses all four of the temporal kernels. For larger Δt, the WE and TimeDiscrete kernels are found to be useful in improving prediction accuracy.

A considerable improvement can be observed after adding spatial information to the model, especially for the prediction of the next few hours. For example, the accuracy of predicting location in 1 h increases from 0.72 to 0.83 by adding the Place kernel.

Besides the two simplistic baseline versions, namely MostPopular and SamePlace, we also compare our proposed method with spatio–temporal Markov model [11], called MarkovCDF, which is considered to be among the most accurate prediction methods for this class of prediction problems [14]. The method sequentially predicts the next destination and the arrival time by combining the transition probability between places and the distribution of visit and transition durations. As the result of a 3rd order model is slightly worse than the 2nd order model, we only report the results of the 1st and 2nd order Markov model, noted MarkovCDF(1) and MarkovCDF(2) respectively.

Please cite this article in press as: T.M.T. Do, et al., A probabilistic kernel method for human mobility prediction with smartphones, Pervasive and Mobile Computing (2014), http://dx.doi.org/10.1016/j.pmcj.2014.09.001

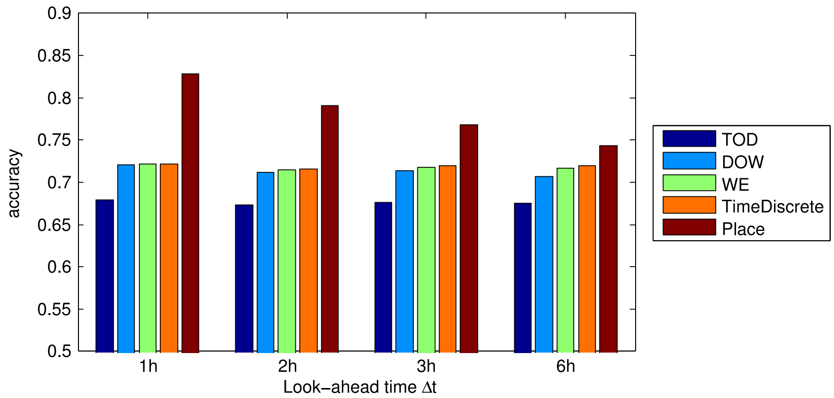


Fig. 6. Accuracy of the spatio-temporal model with increasing number of kernels. The legend should be read as accumulated. Blue corresponds to the model with the TOD kernel only. Dark red corresponds to the model with all the five kernels. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

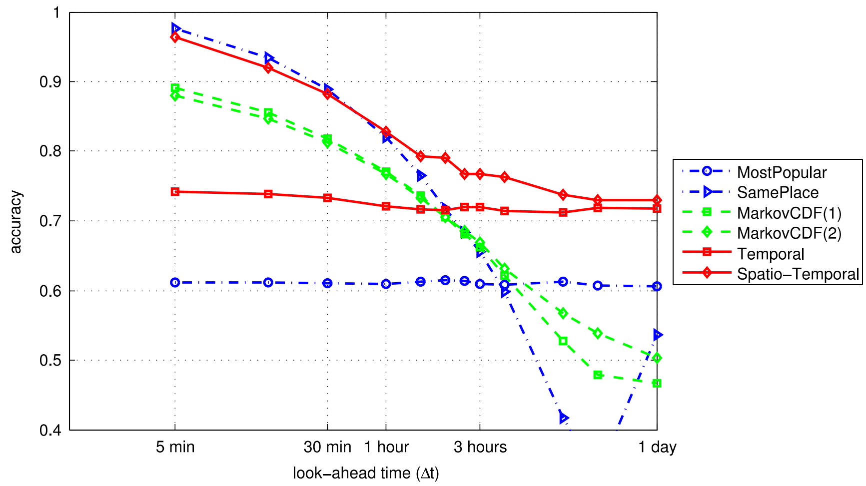


Fig. 7. Comparing kernel-based models with baselines on the WiFi dataset. The Temporal curve corresponds to the model that includes the kernels on temporal information (TOD, DOW, WE, TimeDiscrete). The spatio-temporal model corresponds to the model with kernels on all computed features.

Fig. 7 shows comparative results of our kernel method, called Spatio-Temporal, and the baseline methods. The Temporal method corresponds to our method with only temporal kernels. As can be seen, both the Temporal and the Spatio-Temporal models significantly outperform both the simplistic baseline MostPopular and the more sophisticated baseline MarkovCDF, for all look-ahead time Δt. These results can be explained by two key advantages of the proposed method over the MarkovCDF. First, while the MarkovCDF method only uses the history data points with the exact match of current context, our method defines soft-matching scores between context vectors via the kernel function, allowing us to exploit the historical data more efficiently. Second, the MarkovCDF needs to sequentially fill the location trace from t to $t + \Delta t$ by predicting next location and arrival time. In this greedy approach, a single prediction error will be propagated to subsequent predictions. Our approach instead provides a direct estimate of the conditional probability of a location at time $t + \Delta t$, thus avoiding the propagation of errors in greedy methods [11,14].

Finally, we see that the performance of the spatio-temporal option is generally better than the one of SamePlace method. However, the SamePlace method is still better than the Spatio-Temporal model for the look-ahead time of less than 1 h. This reflects the fact that our model does not completely capture the same place probability distribution. This observation suggests that we could improve the spatio-temporal model by combining it with the SamePlace method.

7. Combining spatio-temporal model with SamePlace model

To combine SamePlace with our kernel method, we first introduce a probabilistic model for the SamePlace method, then employ a convex combination of probabilities as follows:

$$P(l|\mathbf{c}, t, \Delta t) = \alpha_{t,\Delta t} P_{st}(l|\mathbf{c}) + (1 - \alpha_{t,\Delta t}) P_{sp}(l|t, \Delta t), \tag{4}$$

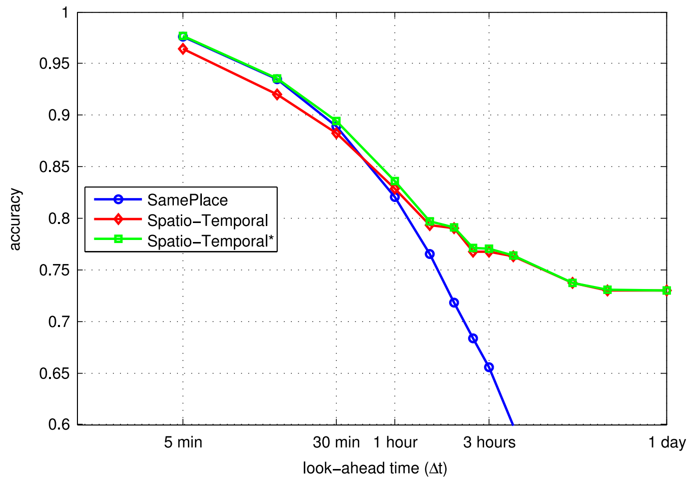


Fig. 8. Overall accuracies of the combined model Spatio-Temporal*, compared to the SamePlace baseline and the Spatio-Temporal model.

where $P_{st}(.)$ is the probability estimated by the Spatio-Temporal model and $P_{sp}(l|t, \Delta t)$ is the probability given by the probabilistic SamePlace model which we will detail in the next paragraph. By combining the Spatio-Temporal model with the SamePlace model, we expect the two models to complement each other in different situations. Importantly, the convex combination of two distributions results in a probabilistic distribution without any additional adaptation. The model remains in a probabilistic framework.

7.1. Probabilistic SamePlace model

Our goal is to model the probability that the user will be at the same place for a given Δt : $P_{sp}(l|t, \Delta t) \triangleq P(l(t + \Delta t) = l(t))$ where $l(t)$ denotes the location at time t. Without any contextual information, this probability coincides with the curve in Fig. 5 if there is no missing data. However, the probability of being at the same place after Δt depends on many factors. We implemented a simple method which uses the time of day as a conditional variable. The intuition is that the probability of being at the same place after e.g., 3 h would be high for night timeslots (e.g., 3 am) while it would be low for the early morning (e.g. 7 am). At the end, for each hour of the day h and each Δt, we have an estimate of $\hat{p}(h, \Delta t)$ for the probability of being at the same place after a time interval Δt given the current hour h. Finally, the probability of being at a place l at time $t + \Delta t$ with the probabilistic same-place model is defined as follows:

$$P_{sp}(l|t, \Delta t) = \begin{cases} \hat{p}(h(t), \Delta t) & \text{if } l = l(t) \text{ and } l(t) \neq OFF \\ \dfrac{1 - \hat{p}(h(t), \Delta t)}{M - 1} & \text{if } l \neq l(t) \text{ and } l(t) \neq OFF \\ \dfrac{1}{M} & \text{if } l(t) \neq OFF, \end{cases} \tag{5}$$

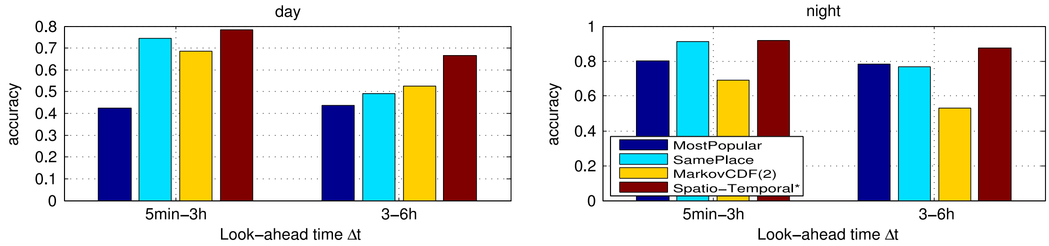
where $h(t)$ is the hour at the prediction time t, M is the number of visited places in the past, and $OFF = -2$ is the special code for missing data.

The probabilistic same-place model can be learned for each individual or for a population. In our implementation, we use data from all users to estimate a general model for the LDCC population. This model performs well in general, but can be inaccurate for outliers (e.g., a person that works a night shift). We leave as future work the implementation of a personalized probabilistic same-place model.

For the combination weight in Eq. (4), we also found that the optimal value for α varies in function of the time of day and Δt. To maximize the performance of the combination model, we optimize $\alpha_{t,\Delta t}$ for each combination of time of day and Δt. The time of the day was divided uniformly into 12 intervals, and the duration of Δt is divided into 4 intervals: 0–30 min, 30 min–1 h, 1–2 h, and more than 2 h. Similarly to kernel parameters, the values of α for each case is optimized on the training data.

7.2. Combination results

Fig. 8 shows the results obtained with the proposed combined model, denoted as Spatio-Temporal*. As can be seen, the combined model improves the accuracies of Spatio-Temporal models for Δt less than 1 h, and slightly outperforms the SamePlace baseline in term of accuracies. It is also interesting to see that the combined method provides some improvements



(a) Daytime. (b) Night period.

Fig. 9. Comparison of prediction accuracy of methods at two periods of the day: 6 am–6 pm (daytime), 6 pm–6 am (nighttime), and various look-ahead times Δt.

for Δt larger than 1 h, although the difference is small. This is not a surprise since the same-place probability is relatively low for $\Delta t > 3$ h, and so becomes less useful for the prediction of future location.

Fig. 9 illustrates the effect of the prediction time on the accuracies of the prediction methods. As human mobility is highly predictable for the night period, we have good accuracies even with baseline methods that do not exploit any contextual information (also see Fig. 4). The prediction accuracies for night periods can be improved by exploiting contextual information (in the spatio-temporal and combined models). However, these improvements are modest compared to the larger improvements obtained for daytime prediction, in which human mobility is more complex.

Considering location prediction in the next three hours, we see that the accuracies of the baseline SamePlace model is high (for daytime prediction the accuracy is 0.74), and the prediction accuracies cannot be improved much by the contextual model. Any improvement in accuracies probably comes from the low fraction of time in which people move within three hours. These periods, however, are interesting from an applicative view point. When Δt increases, the differences between the combined model and the SamePlace baseline are more significant since the performance of this baseline degrades quickly. Our methods perform better than baseline methods in many situations.

The comparative analysis in this section shows that our methods perform better, in many situations, than the baseline methods. While the SamePlace model can be accurate, the prediction is not interesting in practice. This finding indicates that the accuracy measure is not enough for the evaluation of predictors. In the next section, we consider a more applicative prediction task with an alternative evaluation measure.

8. Predicting the most likely set of places for a time interval

In many practical applications, such as contextual reminders or recommendations, rather than knowing *one* exact place that will be visited, it might be important to know what *set of places* will be visited in a given time slot. In practice, we visit places using that strategy, e.g., going downtown in the afternoon might imply visiting shops, cafes, offices, with no predetermined or just an approximate order. We explore this task here.

8.1. Prediction task and evaluation measures

In the previous sections, we studied the task of predicting a unique user location at a given time in the future. This task can be generalized to the case of predicting several places that a user will visit within a time interval in the near future. The task is formalized as follows: *At time t, predict the list of places that the user will visit in the time interval* $(t + \Delta t_1, t + \Delta t_2)$ *given the history of place visits up to time t*. This task can be viewed as an information retrieval task if we consider the current context as the query, the list of places as documents, and the list of places that will be visited as relevant documents [25]. The predictor gives a score for each place, which is used to rank the set of places. We use standard information retrieval measures to evaluate the sorted list of places produced by the predictor. Let N be the number of places in the response; the evaluation measures are computed as follows:

- precision at N: the fraction of the top-N places that are actually visited in the time interval $(t + \Delta t_1, t + \Delta t_2)$.
- recall at N: the fraction of visited places that belong to the list of top-N places in the response.
- F-score at N: the harmonic mean of precision at N and recall at N:

$$F = 2 \cdot \frac{precision \cdot recall}{precision + recall}.$$

8.2. Adapting predictors to the new task

Our proposed model can be easily adapted to perform this task by aggregating prediction results from multiple predictions with various $\Delta t \in (\Delta t_1, \Delta t_2)$. We define

$$score(l|t, \Delta t_1, \Delta t_2) = \sum_{\Delta t \in (\Delta t_1, \Delta t_2)} P(l|t, \Delta t), \quad (6)$$

Please cite this article in press as: T.M.T. Do, et al., A probabilistic kernel method for human mobility prediction with smartphones, Pervasive and Mobile Computing (2014), http://dx.doi.org/10.1016/j.pmcj.2014.09.001

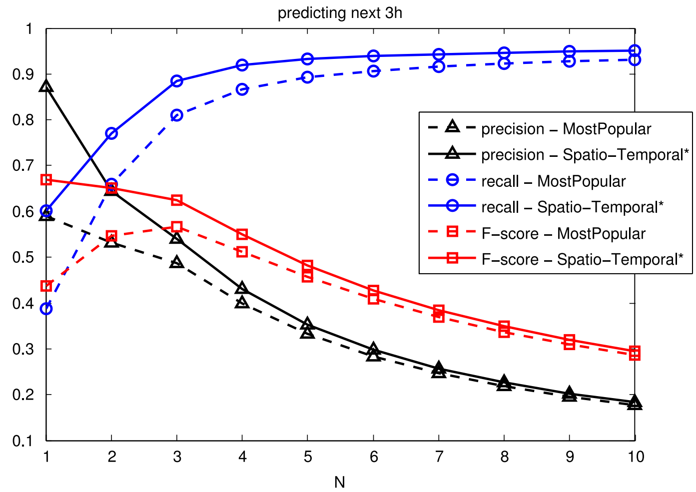


Fig. 10. Precision, recall, and F-score values for predicting the set of N most likely locations for the next 3 h.

where $score(l|t, \Delta t_1, \Delta t_2)$ is the score defining how likely the place l is to be visited during the time interval $(t+\Delta t_1, t+\Delta t_2)$. The list of most likely places can be obtained by ordering the scores in descending order. Note that in the above formula, we use the sum as the aggregation operator but other aggregation operators can also be appropriate depending on the application. For example, if the chronological order is critical then we can consider a weighted sum of probabilities where the weight is inversely proportional to Δt.

Similarly, we can adapt the MostPopular baseline model for this prediction task by using as probability of a given place the popularity (in terms of total stay time) of that place in the history of visits. Since the output of this model is invariant with respect to Δt, there is no need to use the aggregation operator. MostPopular is the only baseline method that we study for this task since the SamePlace method and the MarkovCDF methods are not suitable (they do not output scores over the list of places).

8.3. Experimental setting

To focus on a realistic application, we evaluate the prediction results on daytime predictions. Two settings of the time interval were used; the first one is to make a prediction for the next 3 h ($\Delta t_1 = 0$, $\Delta t_2 = 3$ h), and the second setting is to make a prediction for the time interval from 3 to 6 h ($\Delta t_1 = 3$, $\Delta t_2 = 6$ h). There are four possible predictions per day, uniformly distributed from 8:00 to 17:00 (that is, one prediction every 3 h). In the case of large proportion of missing data (higher than 30%) in the prediction time interval $(t + \Delta t_1, t + \Delta t_2)$, the predicted results are not included in the evaluation due to incompleteness of ground truth.

8.4. Results on prediction location for a time interval

Fig. 10 shows experimental results for the task of predicting user location for the next 3 h (that is, $\Delta t_1 = 0$ and $\Delta t_2 = 3$ h). Recall that we consider four daily prediction times (at 8:00, 11:00, 14:00, and 17:00), which correspond to the four prediction time intervals: 8:00–11:00, 11:00–14:00, 14:00–17:00, and 17:00–20:00. Two models are compared: the non-contextual approach with the MostPopular model (Section 5), and the contextual approach with the Spatio-Temporal* model from Section 7. We compare the performance of the two models by using the three aforementioned evaluation measures with N ranging from 1 to 10. As can be seen, the Spatio-Temporal* model systematically outperforms the baseline model for all evaluation measures and all values of N. As people generally do not visit too many places within 3 h, the precision value drops quickly as N increases. The improvement over the baseline results is also larger for small values of N. For example, the absolute improvements in top-1, top-3, and top-5 F-score are 0.23, 0.04, and 0.03 respectively.

To study the effect of look-ahead time in the performance, we shift the prediction time interval by 3 h ($\Delta t_1 = 3$ h, $\Delta t_2 = 6$ h). Results are shown in Fig. 11. Note that the 4 daily prediction time intervals become: 11:00–14:00, 14:00–17:00, 17:00–20:00, and 20:00–23:00. The final results with the second setting follow similar trends to those observed in the first setting, but the absolute values of precision, recall, and F-score are lower. On one hand, the prediction is more difficult because the look-ahead time is increased. On the other hand, the prediction time intervals are shifted towards the evening period, where the human mobility is more predictable. Note also that these two effects reduce the improvement of the Spatio-Temporal* model over the baseline in term of accuracy (see Fig. 9), which explains why the improvement in this setting is lower than in the first setting (i.e., predicting the next 3 h).

It is also relevant to study the prediction performance on different times of the day. We report these results in Fig. 12. In term of F-score, the predictability is highest at 14:00 (best F-score = 0.72 for $N = 1$) and lowest at 17:00 (best F-score = 0.66

Please cite this article in press as: T.M.T. Do, et al., A probabilistic kernel method for human mobility prediction with smartphones, Pervasive and Mobile Computing (2014), http://dx.doi.org/10.1016/j.pmcj.2014.09.001

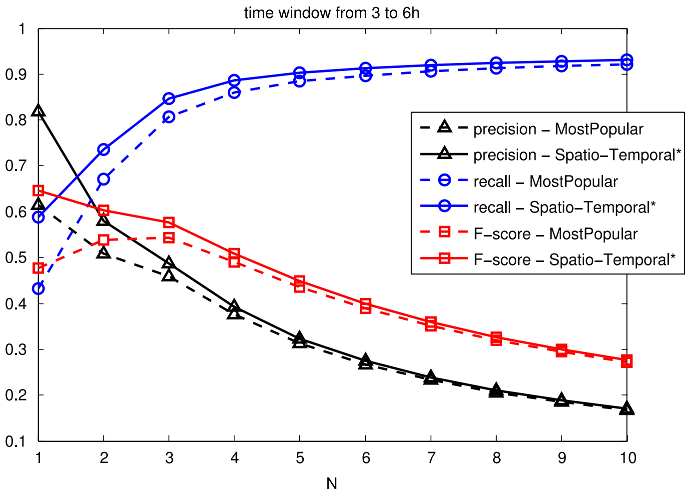
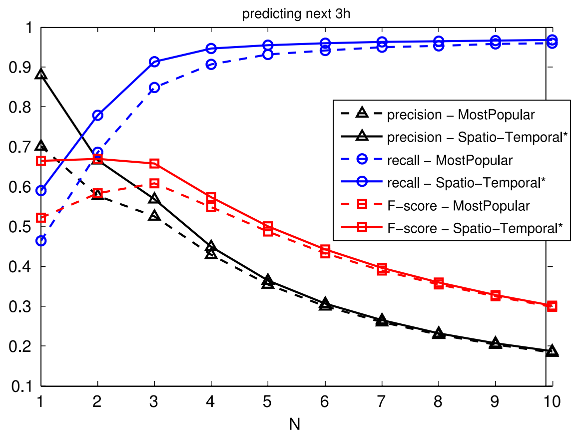
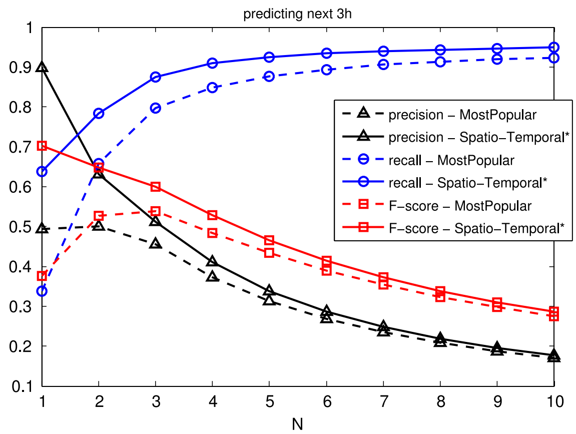


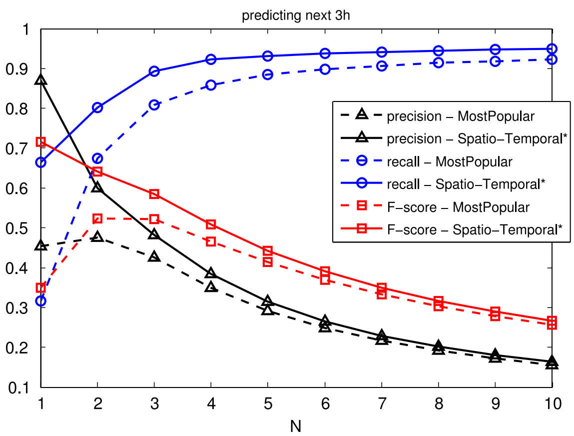
Fig. 11. Precision, recall, and F-score values for predicting the set of N most likely locations for the time window from 3 to 6 h in the future.



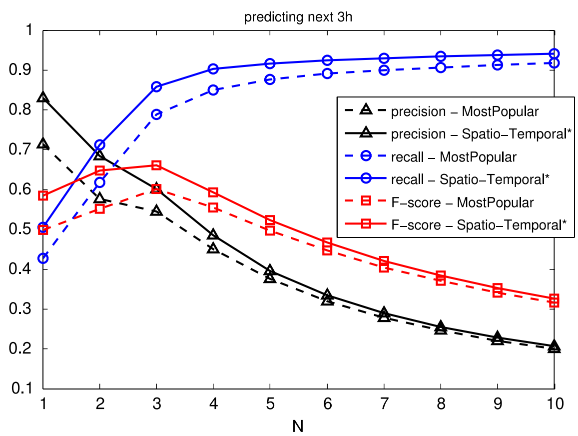
(a) Prediction at 8:00.



(b) Prediction at 11:00.



(c) Prediction at 14:00.



(d) Prediction at 17:00.

Fig. 12. Precision and recall value for predicting the set of N most likely locations for the next 3 h at different time of the day.

for $N = 3$) and 11:00 (best F-score = 0.67 for $N = 2$). This reflects the fact that people usually stay in the same place between 14:00 and 17:00 (e.g., working in an office), so that the response for a single place gets the best F-score. At lunch time 11:00–14:00, the best result is obtained for $N = 2$ since people usually go out for lunch, then come back; however, it is

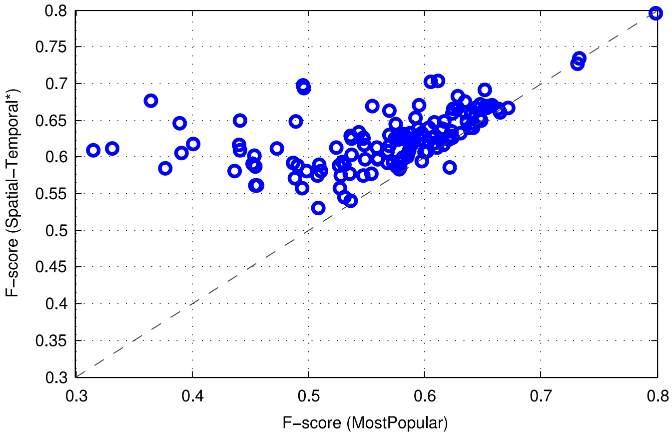


Fig. 13. Top-3 F-scores of the prediction of user location in the next 3 h with MostPopular model and Spatio-Temporal* model. Each circle corresponds to a user in the LDCC data.

not easy to predict exactly where user will go among multiple places. Predictions at 8:00 in the morning (best F-score = 0.7, $N = 3$) are generally more accurate than predictions at 17:00 (best F-score = 0.66, $N = 3$). While the two prediction time intervals correspond to commuting transition periods, the morning transition is more predictable than the late-afternoon transition since people can go to different places (e.g., shopping, bars) before returning home. We also observe that the improvement of contextual prediction over non-contextual prediction is highly correlated with the predictability. The contextual information is most helpful for the difficult prediction time interval starting at 11:00.

Finally, we show how predictability varies among users. The scatter plot of top-3 F-scores is presented in Fig. 13 which shows that the prediction performance is improved with the Spatio-Temporal* model for most users (above the 45° line). Moreover, the improvement is large for users having low F-score with the baseline method. Importantly, the variance in performance for the combined method is smaller than the one for the baseline. Finally, the F-scores seem to follow the normal distribution with mean 0.62 and standard deviation 0.04.

9. Conclusion

We examined the location prediction problem with a new angle by redefining the prediction task. We proposed a probabilistic kernel method for learning the dependency between user location and multivariate contextual variables from sparse data. This is the first study, to our knowledge, that uses the kernel density estimation on multiple temporal and spatial variables for location prediction. To improve the prediction performance, we combined the spatio-temporal model with a baseline model, resulting in a robust probabilistic model which outperforms baseline methods in different situations. Our experiments showed promising results for practical applications of the prediction method, with 84% accuracy for the location prediction of the next hour, and 77% accuracy for the next three hours. For the tasks of inferring the most likely set of places for a given interval, our method reaches an accuracy of 93% with $N = 5$ location candidates for the next 3 hours.

A preliminary version of the method has been implemented on the phone and we did not have any major practical issues. Kernel bandwidths are estimated once, offline, and no more parameter estimation is required since the method is non-parametric. This is an advantage compared to the parametric approach where model parameters need to be updated regularly as the train dataset grows. One limitation of the model is that the required memory and computational resources grows linearly with time. In practice, we can define an upper bound on the size of the data where old data points (e.g., more than 3 months) are automatically removed. Other strategies of data elimination which consider the redundancy are also interesting; we will leave this direction for further studies.

As future work, we are interested in the modeling of trajectories which can improve the prediction performance when a user is on the move. Another direction is to integrate social context to the prediction algorithm by using social sensors and/or considering collaborative prediction tasks for multiple users.

Acknowledgments

Trinh Minh Tri Do and Daniel Gatica-Perez acknowledge the support of Nokia Research through the LS-CONTEXT project. Markus Miettinen and Olivier Dousse were with NRC-Lausanne when this research was done. We thank Juha Laurila and Jan Blom for discussions.

References

[1] J. Laurila, D. Gatica-Perez, I. Aad, J. Blom, O. Bornet, T. Do, O. Dousse, J. Eberle, M. Miettinen, The mobile data challenge: big data for mobile computing research, in: Proc. MDC Workshop, 2012.
[2] K. Mo, B. Tan, E. Zhong, Q. Yang, Your phone understands you, in: Proc. MDC Workshop, 2012.

[3] G. Chittaranjan, J. Blom, D. Gatica-Perez, Mining large-scale smartphone data for personality studies, Pers. Ubiquitous Comput. (2012) 1–18.
[4] J. Hightower, S. Consolvo, A. LaMarca, I. Smith, J. Hughes, Learning and recognizing the places we go, in: Ubiquitous Computing, Vol. 3660, Springer, Berlin, Heidelberg, 2005, pp. 159–176.
[5] D. Ashbrook, T. Starner, Using GPS to learn significant locations and predict movement across multiple users, Pers. Ubiquitous Comput. 7 (2003) 275–286.
[6] T. Liu, P. Bahl, I. Chlamtac, Mobility modeling, location tracking, and trajectory prediction in wireless ATM networks, Sel. Areas Commun. 16 (6) (1998) 922–936.
[7] R. Bajaj, S.L. Ranaweera, D.P. Agrawal, GPS: location-tracking technology, Computer 35 (4) (2002) 92–94.
[8] C. Song, Z. Qu, N. Blumm, A.-L. Barabási, Limits of predictability in human mobility, Science 327 (5968) (2010) 1018–1021.
[9] B. Jensen, J. Larsen, K. Jensen, J. Larsen, L. Hansen, Estimating human predictability from mobile sensor data, in: MLSP, 2010, pp. 196–201.
[10] M. Lin, W. Hsu, Z. Lee, Predictability of individuals' mobility with high-resolution positioning data, in: Proc. UbiComp, 2012, pp. 381–390.
[11] L. Song, D. Kotz, R. Jain, X. He, Evaluating next-cell predictors with extensive Wi-Fi mobility data, Trans. Mob. Comput. (2004) 1414–1424.
[12] J. Krumm, E. Horvitz, Predestination: inferring destinations from partial trajectories, in: Proc. UbiComp, 2006, 243–260.
[13] J. Krumm, A.J.B. Brush, Learning time-based presence probabilities, in: Proc. Pervasive, 2011, pp. 79–96.
[14] S. Scellato, M. Musolesi, C. Mascolo, V. Latora, A.T. Campbell, NextPlace: a spatio-temporal prediction framework for pervasive systems, in: Proc. Pervasive, Berlin, Heidelberg, 2011, pp. 152–169.
[15] T.M.T. Do, D. Gatica-Perez, Contextual conditional models for smartphone-based human mobility prediction, in: Proc. UbiComp, 2012.
[16] V. Etter, M. Kafsi, E. Kazemi, Been there, done that: What your mobility traces reveal about your behavior, in: Proc. MDC Workshop, 2012.
[17] J. Wang, B. Prabhala, Periodicity based next place prediction, in: Proc. MDC Workshop, 2012.
[18] H. Gao, J. Tang, H. Liu, Mobile location prediction in spatio-temporal context, in: Proc. MDC Workshop, 2012.
[19] E. Horvitz, J. Krumm, Some help on the way: Opportunistic routing under uncertainty, in: Proc. UbiComp, 2012.
[20] E. Miluzzo, N. Lane, S. Eisenman, A. Campbell, CenceMe–injecting sensing presence into social networking applications, in: Smart Sensing and Context, 2007, pp. 1–28.
[21] N. Kiukkonen, J. Blom, O. Dousse, D. Gatica-Perez, J. Laurila, Towards rich mobile phone datasets: Lausanne data collection campaign, in: Proc. ICPS, Berlin, 2010.
[22] V.W. Zheng, Y. Zheng, X. Xie, Q. Yang, Collaborative location and activity recommendations with GPS history data, in: Proc. WWW, ACM, 2010.
[23] T.M.T. Do, D. Gatica-Perez, The places of our lives: visiting patterns and automatic labeling from longitudinal smartphone data, Trans. Mob. Comput. 13 (3) (2014) 638–648.
[24] A. Peddemors, H. Eertink, I. Niemegeers, Predicting mobility events on personal devices, Pervasive Mob. Comput. 6 (2010) 401–423.
[25] R. Baeza-Yates, B. Ribeiro-Neto, et al., Modern Information Retrieval, Vol. 463, ACM Press, New York, 1999.