Pervasive and Mobile Computing 9 (2013) 516–527

Contents lists available at SciVerse ScienceDirect

Pervasive and Mobile Computing

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

Classification and prediction of whereabouts patterns from the Reality Mining dataset

Laura Ferrari, Marco Mamei *

Dipartimento di Scienze e Metodi dell'Ingegneria, University of Modena and Reggio Emilia, Italy

ARTICLE INFO

Article history:
Received 31 December 2010
Received in revised form 16 December 2011
Accepted 15 April 2012
Available online 25 April 2012

Keywords:
Whereabouts data
Human mobility
LDA topic extraction
Topic classification
Topic prediction

ABSTRACT

Classification and prediction of users' whereabouts patterns is important for many emerging ubiquitous computing applications. Latent Dirichlet Allocation (LDA) is a powerful mechanism to extract recurrent behaviors and high-level patterns (called *topics*) from mobility data in an unsupervised manner. One drawback of LDA is that it is difficult to give meaningful and usable labels to the extracted topics. We present a methodology to automatically classify the topic with meaningful labels so as to support their use in applications. We also present a topic prediction mechanism to infer user's future whereabouts on the basis of the extracted topics. Both these two mechanisms are tested and evaluated using the Reality Mining dataset consisting of a large set of continuous data on human behavior.

1. Introduction

The diffusion of smart phones equipped with localization capabilities allows to collect data about mobility and whereabouts from a large user population in an economically feasible and unobtrusive way [1–3]. This information opens new scenarios and possibilities in the development of context-aware services and applications, but several challenges need to be tackled to extract practically useful information from such mobility datasets [4].

The Reality Mining dataset [5] is a seminal dataset in this area. It collects data about the daily life of 97 users over 10 months. One key part of this dataset is the log of places visited by the users, identified via GSM-based localization. Some pioneering researches started to apply pattern-analysis and data mining algorithms to such mobility dataset. In particular, the work described in [6] applies Latent Dirichlet Allocation (LDA) [7] to extract high-level patterns (called *topics*) from mobility data—see next Sections for a description of this approach.

In this paper we build on top of such an LDA model and provide two key contributions:

The first contribution is to present a method to automatically classify the patterns extracted by the LDA algorithm. Once meaningful labels are given, the extracted patterns become readily understandable and usable in applications. For example, a life-logger application [8] could readily use the extracted label to automatically create an entry in the user blog. Similarly, analyzing city-wide mobility patterns, applications could identify routine behaviors affecting city-life and communicate such information to local government and city planners [9]. These tasks are simplified once a proper label is assigned to the discovered patterns, while they are very difficult starting from the raw output of the LDA algorithm.

The second contribution of this paper is to present a novel methodology to predict future users' whereabouts on the basis of LDA topics and to carefully assess the obtained predictions in different settings. LDA techniques are useful also in predicting future whereabouts, in that the representation of a day in terms of its topics highlights the actual underlying patterns; it is particularly resilient to noise, and thus it allows to predict future evolution of the users' whereabouts. Several useful

* Corresponding author.
E-mail address: marco.mamei@unimore.it (M. Mamei).

1574-1192/$ – see front matter © 2012 Elsevier B.V. All rights reserved.
doi:10.1016/j.pmcj.2012.04.002

applications naturally arise given the possibility of predicting where the user is likely to go in the future [10]. For example, advanced memory-aid applications [11] could be extended to alert the user about likely future events (e.g., "you will miss your meeting, if you do not move now") or to better contextualize reminder notifications.

The combination of the proposed classification and prediction is a powerful tool in that it allows to express what is going to happen to the user (i.e., which topic is going to be expressed) with high-level meaningful labels. In particular, our contributions can notably support and simplify the adoption of LDA results to ubiquitous computing services and applications. We want to stress the fact that we are not trying to extend the LDA model, our classification procedure works on top of that, and can be generalized to work on top of other models with similar outputs such as those presented in [5].

The remainder of this paper is organized as follows: we first describe background and motivations for the proposed contributions and survey some related work in the area. Then we discuss data preprocessing and the LDA algorithm that are at the basis of our work. The following two sections describe our approach to associate meaningful labels to the LDA-topics, and our algorithm to predict user's future whereabouts. Eventually we draw some conclusions and future work.

2. Background and motivations

In recent years several researches applied pattern analysis and data mining algorithms to extract high-level information and routine behaviors from people mobility dataset.

A number of these researches focus on 2 "similar" techniques: Principal Component Analysis (PCA) [12,5] and Latent Dirichlet Allocation (LDA) [7,6]. The goal of both these techniques is to discover significant patterns and features from the input data. More precisely, from a maximum-likelihood perspective, both these techniques aim at identifying a set of latent variables z and conditional probability distributions $p(x|z)$ for the observed variable x representing users' whereabouts. The latent variables z are typically of a much lower dimensionality than x. Thus they encode patterns in a more understandable way with reduced noise.[1] These techniques have been applied to a variety of people mobility datasets [5,6,13] with similar modalities.

In the context of the Reality Mining dataset (that is also the target of our work), the approach consists of extracting for each user and for each day a 24-slot array indicating where the user was at a given time of the day (24 h). User's locations are expressed as either: 'Home','Work','Elsewhere' or 'No Signal', the latter indicating lack of data (in the remainder of this paper we refer to these locations respectively as '*H*', '*W*','*E*','*N*'). So, for example, a typical day of a user could be '*HHHHHHHHHWWWWWWWWWEEEHHH*' expressing that the user was at home at night and early morning, then went to work until late afternoon, then went to somewhere else for three hours, and finally went back home. Despite the coarse grain labeling of places ('*H*','*W*','*E*','*N*'), several interesting patterns can be identified. They include: "stay out late at night" – transition between '*E*' and '*H*' happening late at night, and "arrive late at work" – '*W*' place identified later than usual.

Applying PCA or LDA to a set of these arrays allows to extract some low-dimension latent variables (eigenvectors and LDA-topics respectively) representing underlying patterns in the data, and offering conditional probability distributions for the observed arrays (i.e., days). Fig. 1 illustrates some eigenvectors and LDA-topics extracted from the Reality Mining dataset.

Eigenvectors, in Fig. 1.a, encode the probability of the user being at a given location: '*H*', '*W*','*E*'. Similarly, LDA-topics encode the probability of the user being at a given location (in a different representation format—see next Section for details). Fig. 1.b shows the most probable days according to the conditional probability distribution of a given topic. These days are thus a representation of the topic itself.

Assigning a *meaning* to the extracted latent variables is a difficult task, that has been typically performed by visually inspecting the latent variable itself or the days that strongly correlate to the variable (i.e., most probable days given the latent variable) [5,6,13]. For example in [5], by visually inspecting the first eigenbehavior represented in Fig. 1.a, authors conclude that it relates to the typical working day routine behavior. The second eigenbehavior corresponds to typical weekend behavior. Similarly, Fig. 1.b reports some LDA-topics obtained in [6]. By visually inspecting the days strongly correlated to the extracted topics, authors conclude that topic 20 means "at home in the morning" while topic 46 means "at work in the afternoon until late in the evening". Looking at these examples, it is clear how difficult it is to give a meaning to the extracted patterns and how difficult it is to evaluate the quality of the given meaning (i.e., label). As illustrated in Fig. 1 the task of labeling a given pattern can be even more difficult than labeling individual days in that the labels for patterns should meaningfully describe a whole set of days.

Our aim is to solve this issue by proposing a mechanism to classify the extracted latent variables. In particular we are not trying to label individual days, but the whole routine pattern. First pattern analysis "clusters" similar days together, then our approach gives a label to the whole cluster.

The need to associate meaningful labels to topics have been also considered in text-mining applications. The work presented in [14] classifies latent topics extracted from text corpora. Although this work applies to a completely different scenario, the fact that topic understanding is an important research challenge also in other communities further motivates our work.

[1] The traditional definition of PCA does not involve maximum-likelihood computation, but only the eigenvector-decomposition of the covariance matrix of the dataset. PCA based on maximum-likelihood is sometimes referred as probabilistic-PCA [12].

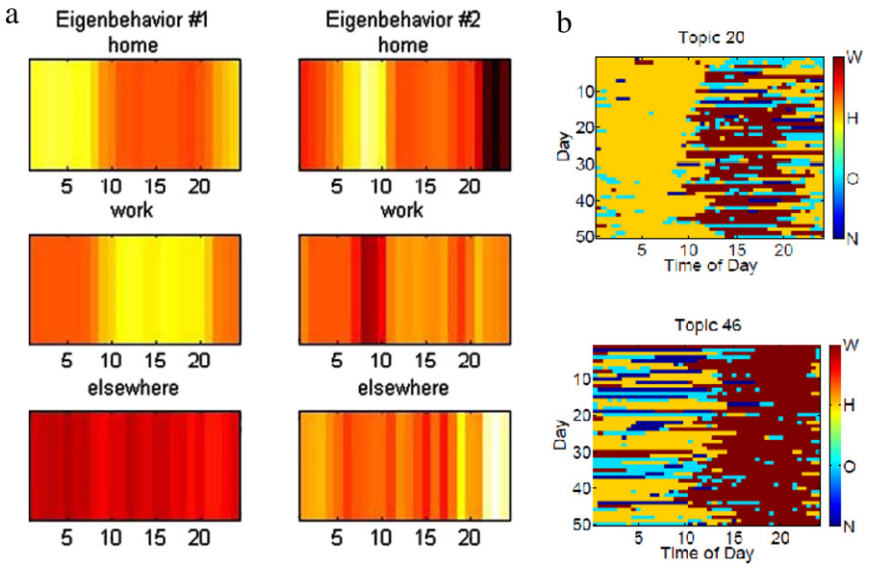


Fig. 1. (a) The top two eigenbehaviors for Subject 4 of the Reality Mining dataset, the lighter the color the higher the probability. (b) Exemplary LDA-topics extracted from the Reality Mining dataset. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)
Source: Image (a) taken from [5] and image (b) taken from [6].

Despite the fact that the presented approaches are generalizable both to PCA and LDA, in the following of this paper we focus only on the LDA application. The probabilistic model realized by LDA is better suited at extracting different patterns from complex datasets [7,6]. LDA has two main characteristics that make it suitable to our pattern discovery task. On the one hand, it is an unsupervised approach: it does not require to define classes (i.e. topics) *a priori* and it does not require difficult-to-be-acquired labeled data. On the other hand, topics represent meaningful probabilistic distributions over words and documents. This allows to analyze and understand the routine behavior they stand for.

3. Related work

The recognition of location as a primary source of context information has stimulated a wealth of work. In particular, the core problem addressed in this paper: understanding and predicting users' whereabouts is present in several related works.

3.1. Identifying places

Several researches tackle the problem of understanding people whereabouts by trying to extract and identify those places that matter to the user. Mainstream approaches are either based on segmenting and clustering GPS-traces to infer what are the places relevant to the user [15–17], or on detecting places and mobility on the basis of nearby RF-beacons such as WiFi and GSM towers [18,19,5,20]. These approaches require the user to run a special software on her device to collect and analyze the log of GPS or RF-beacons available. Thus experiments with these mechanisms are usually conducted with a relatively small user population (the Reality Mining dataset used also in this paper is by far one of largest datasets in this category). Tracking user location on a larger-scale typically involves the use of a log of the GSM cell tower being visited provided by a Telecom operator, or geo-tagged user-generated content from e.g., Flickr and Twitter, that implicitly denotes user location.

In summary, these approaches allow to represent and understand users' mobility patterns as a sequence of places being visited at different times of the day. This representation resembles the "Home, Work, Elsewhere, No Signal"-representation used in the Reality Mining dataset.

Accordingly, while this paper focuses on higher-level abstractions (the sequence of places being visited is our starting point), the above approaches are the fundamental elements to apply our algorithms to other mobility datasets.

3.2. Identifying and predicting routes

A number of related work deals with the problem of identifying and predict the routes the user takes to move from one place to another. These approaches run data mining algorithms to identify recurrent patterns in the GPS tracks from multiple users. Works in this area can be divided in 2 broad categories: "geometric-based" approaches [21] apply pattern matching to the sequence of geographical coordinates composing the tracks. We call it geometric in that they use the physical "shape" of the path to compute the matching among routes. "String-based" approaches, instead, create a symbolic representation

a

UserID	Begin	End	CellID
22	2004-08-27 14:00	2004-08-27 16:00	102
22	2004-08-27 16:30	2004-08-27 17:00	122
...	...	...	...

b

UserID	CellID	Label
22	102	Home
22	121	Work
...	...	...

Fig. 2. Tables used in the Reality Mining dataset.

of the path (e.g., by considering only the names of the areas crosses by the path) and apply pattern-matching on that list of symbols [22,23]. In both cases, the extracted routes can be both used to classify the user current and past whereabouts, but also to predict the user's next movements.

The work presented in this paper is similar to 'string-based'' approaches, in that the geographic information about the places visited by the users are lost in favor of the more compact "Home, Work, Elsewhere, No Signal''-representation. However, there are two important differences. On the one hand, our representation is even more abstract that the one discussed in [22] and similar works. Labels in our dataset (e.g., Home) are completely detached from their physical location and, in fact, different users will label as "Home'' completely different places. On the other hand our classification algorithm could extend also the above related work to obtain a more descriptive label for the extracted routes.

3.3. Identifying and predicting routines

Even more high-level than extracting places and routes, is the problem of identifying and predicting routine whereabouts behaviors.

The work presented in [13] is based on extremely large anonymized mobility data coming from Telecom operators and has the goal of monitoring and describing the spatio-temporal dynamics of the city. On the one hand, it is possible to identify hot-spots in the city (e.g., popular areas). On the other hand, it is possible to determine and predict the "activity-level'' of a given day by comparing the number of people located in a given area in real time with past measures.

In this context the works that most directly compare to our proposal are [5,6] that we have already introduced in the previous section. They use PCA and LDA algorithms to extract routine behaviors from the Reality Mining dataset. Moreover, they provide prediction mechanisms to identify the routine the user will follow during the day. On the one hand, our work provides a further classification step to give meaningful labels to the extracted routines. Moreover, it proposes a novel classification mechanism that improves the current state of the art.

From a complementary perspective, the work in [24] uses an information-theoretic approach with the aim of quantifying user whereabouts' predictability. Authors analyzes the Reality Mining dataset to identify which time-periods reduce uncertainty on a mobile phone user's routines. It is worth noticing that such work presents results in line with our findings (Section 5).

4. Data preprocessing and LDA

In this section we present the data and algorithms representing the starting point of our work. Most of the mechanisms described in this part are taken from the literature [5,6]. However, some parts like the Support Vector Machine (SVM) used for data preprocessing are original of our work.

4.1. Data preprocessing

As already introduced, the work presented in this paper is based on the GSM-localization part of the Reality Mining dataset. This dataset basically consists of two big tables (see Fig. 2). For each user are recorded several time-frames and the GSM towers where the user was connected (see Fig. 2.a). Tables have missing data (time-frames in which no information has been logged) due to data corruption and powered-off devices. On average logs account for approximatively 85% of the time that the data collection lasted. Another table (see Fig. 2.b) records the labels given by the users to the different GSM towers ("Home", "Work" and other places overall considered "Elsewhere''). Not all the towers are labeled. The dataset comprises 32,628 GSM towers and only 825 are labeled (2.5%). Fortunately labeled cells are those in which users spend most of the time so overall 75% of the dataset happens to be in labeled cells. Still, identifying where the users have been in the remaining 25% of the time is an important issue to improve the data.

In its original format the Reality Mining dataset is thus noisy and with a lot of missing values. Although some works [6] try to extract patterns directly from such data (considering as "Elsewhere'' all the unlabeled towers), we opted to run some preprocessing to complete missing values.

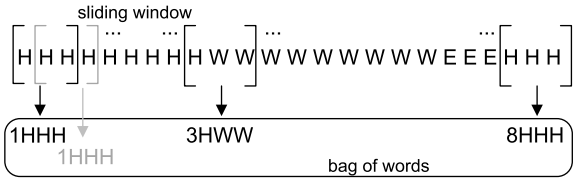


Fig. 3. Sliding windows approach to obtain the bag of words describing user whereabouts. The bag of word does not contain duplicates.

This is also the approach proposed in [5] where an HMM has been used to complete missing information. In this paper we use a SVM instead as it is simple to adopt and provides good classification performances. Our approach consists of training a SVM to infer the labels of all the GSM towers. SVM computations have been performed with the LIBSVM library,[2] on the basis of the following procedure:

1. We create training and testing set as:

Day of week	Weekend	Hour	CellID	Label
Tuesday	no	14	150	Work
Saturday	yes	17	950	Home
Wednesday	no	15	155	?

   The table associates the *label* to be identified to a feature vector consisting of the day of the week when the tower is visited, whether it is a weekend or not, the hour of the visit, and the cell ID.
2. We converted all the data in the training and testing sets in numerical attributes and performed a simple scaling on the data, converting all the values to a [0, 1] scale. This is to avoid attributes in greater numeric ranges dominating the others (we acknowledge that converting in numbers and scaling categorical data is useless from a theoretical point of view, but it is required by the LIBSVM library).
3. Following other examples, we consider Radial Basis Function (RBF) kernel since it well-applies to a vast range of classification problems. In addition, we use cross-validation to find the best parameters C and γ of SVM and RBF respectively. Basically we try all the combinations of the parameters with an exponentially-growing grid search. Parameters producing best cross-validation accuracy are selected for the final model. In order to speed-up the computation, cross validation has been conducted on a reduced training set (randomly selecting feature vectors from the original data).
4. We use the best parameters to train the SVM model on the whole training set and to classify the testing set.

SVM classification produces results with an overall 86% accuracy in cross validation (accuracy being defined as the proportion of correct results in the population). Accordingly, this approach reliably completes the missing slots of the dataset. After SVM-classification, the dataset is much more representative of the *plausible* whereabouts of the users (missing groundtruth information, we cannot make assertions on their *actual* whereabouts). For example, without SVM, a lot of days of the users are described by being always "Elsewhere" (i.e., not at home nor work). This is rather unrealistic and in fact SVM corrects this unbalance by restoring, for example, the being-at-home-at-night behavior.

We stress the fact that such a procedure does not simplify data complexity. It just completes entries with missing information that otherwise would have been discarded or simply completed via the "Elsewhere" label.

Following [5,6], we organize the dataset into a sequence of days each consisting of 24 time-slots lasting 1 h. Each time slot is labeled after the cell where the user spends most of his time. If no information are present for that time slot, the cell is marked as 'No-Signal'.

To apply the LDA algorithm described in the following section, the dataset has been further processed. Each day is divided into a sequence of *words* each representing a 3 h time-slot. A 3-h sliding window runs across the day, each word is composed of an integer value in (1, 8) (we refer to this value as *time-period*) and the 3 ('*H*','*W*','*E*' or '*N*') labels in the sliding window. The *time-period* abstracts the time of the day, it is 1 if the sliding window starts in 0:00–3:00 am, 2 in 3:00–6:00 am, and so on (see Fig. 3).

The fact of using *time-periods* of 3 h each (in contrast with some previous work [6] in which different *time-periods* are skewed) improves the resulting dataset in that the number of words for each time slot is not biased by its length, but better reflects the actual user behavior.

The resulting bag of words summarizes the original dataset and is the input data structure for the LDA algorithm described in the next subsection.

2 http://www.csie.ntu.edu.tw/~cjlin/libsvm.

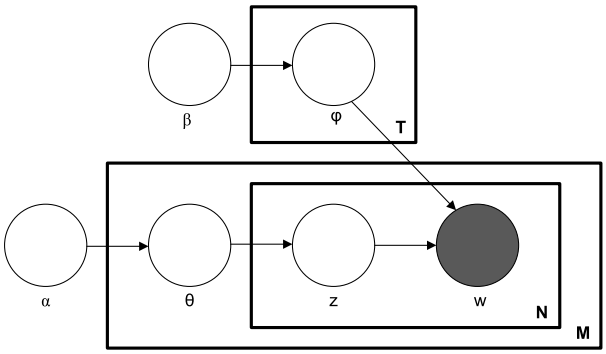


Fig. 4. Plate notation representing the LDA model. α is the parameter of the uniform Dirichlet prior on the per-document topic distributions. β is the parameter of the uniform Dirichlet prior on the per-topic word distribution. θ_i is the topic distribution for document (day) i, ϕ_j is the topic distribution for word j, z_{ij} is the topic for the j-th word in document i, and w_{ij} is the specific word. The w_{ij} are the only observable variables, and the other variables are latent variables.

4.2. LDA algorithm

LDA is a probabilistic generative model [7] used to cluster documents according to the topics (i.e., word patterns) they contain. LDA is an unsupervised learning mechanism that do not require a labeled (difficult to be acquired) training set. Moreover, LDA has two key advantages compared to other clustering mechanisms (such as *k*-means): (i) The LDA model results in probabilistic distributions of days given all topics whereas other clustering algorithms (e.g., *k*-means) assigns only one cluster per day. (ii) Meaningful word distributions as the representation of topics. Topics are based on discriminative location sequences characterizing routines [6].

More in detail, LDA is based on the Bayesian network depicted in Fig. 4. A word w is the basic unit of data, representing user location at a given *time-label*. A set of N words defines a day of the user (i.e. a document). Each user has a dataset consisting of M documents. Each day is viewed as a mixture of topics z, where topics are distributions over words (i.e., each topic can be represented by the list of words associated to the probability $p(w|z)$). For each day i, the probability of a word w_{ij} is given by $p(w_{ij}) = \sum_{t=1}^{T} p(w_{ij}|z_{it})p(z_{it})$, where T is the number of topics. $p(w_{ij}|z_{it})$ and $p(z_{it})$ are assumed to have multinomial distributions. Mixture parameters are assumed to have Dirichlet distributions with hyperparameters α and β respectively. Hyperparameters α and β were set to $\alpha = 50/T$ and $\beta = 0.01$. These are the standard values proposed in the literature [7]. LDA uses the EM-algorithms to learn the model parameters. In our implementation we use the library Mallet (http://mallet.cs.umass.edu) to perform these computations. Once the model parameters have been found, Bayesian deduction allows to extract the topics best describing the routines of a given day (rank z on the basis of $p(d|z)$).

However, as already introduced, since z are just distributions over words, it is difficult to give them an immediate meaning useful in applications.

We want to emphasize that we are not extending the LDA model, we take the model as it is. The next two sections contain the main contributions of this paper: working on top of LDA, we propose mechanisms to give labels to topics z and to predict which topic z the user is in.

5. LDA-topic classification

5.1. Method

In extreme summary, our approach consists of identifying a set of labels describing key aspects of a user typical day. For example, 'Work 9:00–18:00' represents a day in which the user is at work from 9:00 to 18:00. We then identify the LDA-topics representing such a label (we refer to these topics as *label-defined* topics). LDA-topics extracted from the Reality Mining dataset are labeled after the most similar *label-defined* topics. Thus, rather than being described only in terms of probability distributions over words, topics get a meaningful label like 'Work 9:00–18:00'.

Our methodology is based on these key points:

1. We create a set of predefined labels expressing common user patterns, each composed of a place ('*H*', '*W*', '*E*' or '*N*'; we refer to these places as *pattern-label*) and a time-frame representing different parts of the day ('0:00–8:00', '8:00–12:00', '12:00–14:00', '14:00–19:00', '19:00–21:00', '21:00–00:00', '0:00–12:00', '8:00–19:00', '12:00–00:00'; we refer to it as *time-frame-label*). We have 36 labels associated with all possible combinations between the 4 *pattern-labels* and the 9 *time-frame-labels*. For example, the label '*W* 8:00–19:00' represents the pattern where the user is at work from 8:00 to 19:00 while the label '*H* 12:00–14:00' represents the pattern where the user is at home at lunch time. These labels describe the main trends in users' daily routine.

While having 4 *pattern-labels* is dictated by the labels present in the dataset, having 9 *time-frame-labels* is an arbitrary choice to represent meaningful parts of the day. A different set of *time-frame-labels* could be selected to describe different patterns in specific application scenarios. It is also worth noticing that, although we have just 36 labels, the number of routines that we can describe is actually larger in that some patterns can be described by a combination of labels illustrating the user behavior in different parts of the day. For example, a day in which the user went out late at night and (because of that) the next day arrived late at work would be represented by two highly-probable topics labeled after, e.g., 'be out late at night—*E* 00:00–08:00' and 'be late at work—*W* 14:00–19:00'.

2. For each predefined label, we create a set of 15 sample days representing the corresponding daily behavior (each day is represented as a 24-slot array indicating where the user was at a given time of the day). Each day is constructed by filling the time-slots associated with the *time-frame-label* with the corresponding *pattern-label*, and by completing the remaining slots with random *pattern-labels*. The 15 days encode the different nuances in which that particular behavior can be expressed. For example, one sample day associated to the '*W* 8:00–19:00' pattern may express the fact that the user goes Home after 19:00, while another day may express that the user goes Elsewhere after 19:00. The different days keep the pattern constant and change the rest of the day.
3. For each block of 15 days, we compute one LDA-topic. The topic encodes the routine behavior that is most present in the 15 days, thus it encodes the pattern that has been used to generate the days. For example, recalling that topics are distributions over words, the days associated with the '*W* 8:00–19:00' pattern will create a topic in which the $p(w|z)$ of words like 3*WWW*, 4*WWW*, 5*WWW* will be high compared to other words probabilities. The final result is a set of 36 *label-defined* topics, each representing one of the predefined labels. We verified that the resulting topics are not strongly affected if being produced by a number of days smaller or greater than 15.
4. Topics extracted from the Reality Mining dataset are classified using *k*-Nearest Neighbor (*k*NN) in which we use the Kullback–Leibler divergence as distance metric from the above *label-defined* topics.
5. The result is that days of a user can be described as easily-understandable labels (e.g., '*W* 8:00–19:00'), rather than with just probability distributions over words that are more complex to be interpreted.

This kind of labeling scheme has two key advantages. On the one hand, it is *simple*: applications accessing this data can readily understand users' whereabouts by just parsing the topic labels. On the other hand, it allows to express articulated users' routines by combining multiple labels. It is also worth noticing that the approach is extensible. Should another pattern-label be available in the data (e.g., '*P*'-pub), it can be flexibly added to the system to describe topics in more detail.

5.2. Experiments and discussion

We conduct some experiments to test the above approach. First, we experiment with an artificially-created dataset where we get groundtruth information, and thus classification accuracy can be precisely evaluated. The results of these experiments are not directly generalizable to real human patterns and serve for the purpose of testing the correctness of the algorithm. Then, we test the system with the Reality Mining data that misses groundtruth information.

The first set of experiments studies classification accuracy: we want to verify if our approach is able to assign labels to topics in an accurate way. To this end we applied the LDA algorithm to a set of days for which we knew beforehand the actual routine labels. Then we apply our approach to label the resulting topics and verify if the two labels match. More in detail, starting from the above described 36 predefined labels expressing user patterns, we create a testing set on the basis of the following procedure: we randomly pick L labels among the above 36 ones. For each label we create 15 days (following the same procedure described above) and we stack the $15 \cdot L$ days together to create an artificial dataset of user's whereabouts. The L labels represent the groundtruth user's whereabouts patterns.

We extract L topics from this dataset and classify the topics with the *k*NN algorithm (in this experiment with just use $k = 1$). We compute classification accuracy as:

$$\frac{|\{\text{classified labels}\} \cap \{\text{groundtruth label}\}|}{|\{\text{classified labels}\}|}.$$

Results are averaged over 100 runs of the experiment in which we generate random groundtruth topics and random days containing that topics.

To test the *k*NN algorithm in the most direct way, we test with $L = 1$ (testing set composed of one LDA-topic for each predefined label). We obtain a classification accuracy of about 96%. The 4% error mainly comprises misclassified labels representing short *time-frame-label*. For example, if the predefined label is '*H* 12:00–14:00', a testing day could be '*EEEHHHHHWWW***HHH***WWWWWHHHHH*' which is better represented by a topic described by the label '*W* 8:00–19:00' or '*H* 21:00–0:00'.

Testing with $L = 1$ is rather artificial, since realistic user's dataset would comprise different days' routines (i.e., different patterns). Fig. 5 shows the average classification accuracy as a function of the number of patterns used to create the testing dataset. As above mentioned, the loss in the accuracy classification obtained with a little number of patterns is due to the labels representing short *time-frame-label* randomly chosen and not to the variety of the dataset. This fact is also reflected by the higher variation between the minimum and maximum classification accuracy obtained with a small number of patterns.

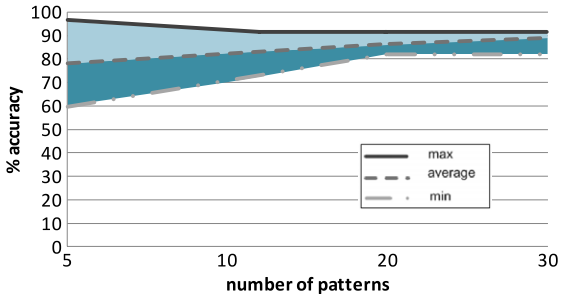


Fig. 5. Average classification accuracy (y-axis) obtained as a function of the number of patterns (x-axis) used to create the testing datasets. The plot reports the minimum, the average and the maximum classification accuracy obtained over 100 runs. Minimum and maximum accuracies give a fair measure of the variance in our results.

In a second group of experiments, we test our classification method on the Reality Mining dataset. We experiment with 36 individuals and 121 consecutive days (from 26 August 2004 to 21 December 2004). We chose this subset of days with the goal of analyzing people and days for which the data is reasonably complete and with the goal of comparing our results with those presented in [25]. It is wroth noticing that also the work in [24] focuses on this subset of data.

Since groundtruth information regarding user topics are not available, our experiments on the Reality Mining dataset focus on two main aspects:

1. We evaluate whether the predefined labels associated to the user's topics are informative enough to reconstruct the day of the user.
2. We evaluate if there are other labels describing user days better than the ones selected by our approach.

With regard to the former aspect, we extract 100 LDA-topics from all the days of each user taken into consideration. For each day d we rank the topic z according to $p(d|z)$, discarding those topics for which $p(d|z) = 0$. For each topic z, starting from the ones with higher $p(d|z)$, we consider the time-frame of the label (*time-frame-label*) assigned to z, and we reconstruct the part of day d associated with that time-frame (e.g., assuming z has label '*H* 8:00–12:00', we reconstruct the 8:00–12:00 time-frame) with the place in the label (*pattern-label* e.g., *H*). If a part of the day has been already reconstructed by a previous (more probable) topic, the mechanisms leaves it unchanged. Parts of the day that do not appear in the considered topics labels are not reconstructed. We then compare the real and the reconstructed day. For each time-slot (hour) we assign an error equals to 1 if the reconstructed label is wrong. While an error equals to 0.5 if that hours is not reconstructed. Fig. 6.a shows the distribution of days reconstruction accuracy. Fig. 6.b shows days reconstruction accuracy as a function of the number of LDA-topics extracted from users' days. The low number of LDA-topics necessary to obtain high accuracy level can be explained by the limited number of users' days available and by their repetitiveness. For the sake of comparison, it is worth noticing that this experiment is analogous to the one in Fig. 5 in the case of real data.

With regard to the latter aspect of the Reality Mining experiments, we extract 100 LDA-topics from all the days of each user taken into consideration. For each day d, we want to find the topic that best describes that day. We rank topics z according to $p(d|z)$. However, rather than considering the first topic in the list as the best, we also take into consideration the length of the *time-frame-label* assigned to the topic. The idea is that topics explaining a bigger part of the user day are to be preferred. In particular, we consider that a topic z_i is better than z_j if it is associated to a bigger *time-frame-label* and $p(d|z_i) > 80\% * p(d|z_j)$.

The most probable topic z_{top} is selected for describing the day. We then evaluate how good the label assigned to z_{top} describes the day. In particular, for each time-slot in the *time-frame-label* we assign an error equals to 1 if the reconstructed label is wrong. For each time-slot that is not in the *time-frame-label* we give an error of 0.5. The idea is to lower the performance of topics associated to short *time-frame-label*, thus describing only a fraction of the day. Finally, we evaluate with the same error measure if there exists another label, better describing the day. We obtain that the labels associated to the selected topics are better than any other labels in the 80% of the cases.

All these results support the use of our classification mechanism.

In summary, the presented topic mechanism allows to attach meaningful labels to topics with limited user involvement. In particular, it is required that users give labels to the places they visit (this task can be simplified by the approaches described in Section 3.1) and that a proper set of *time-frame-labels* are defined. On this basis, the proposed mechanism produces meaningful labels to facilitate applications in taking advantage of the extracted topics.

The described mechanism is the key original contribution of this paper. To the best of our knowledge there are not approaches in the literature that are suitable for a direct comparison. In general, we use simple non-parametric classification mechanisms as they provide good performances without requiring underlying assumptions. Even if obtained results are already rather good, in our future work we will try to improve them further with different classification mechanisms.

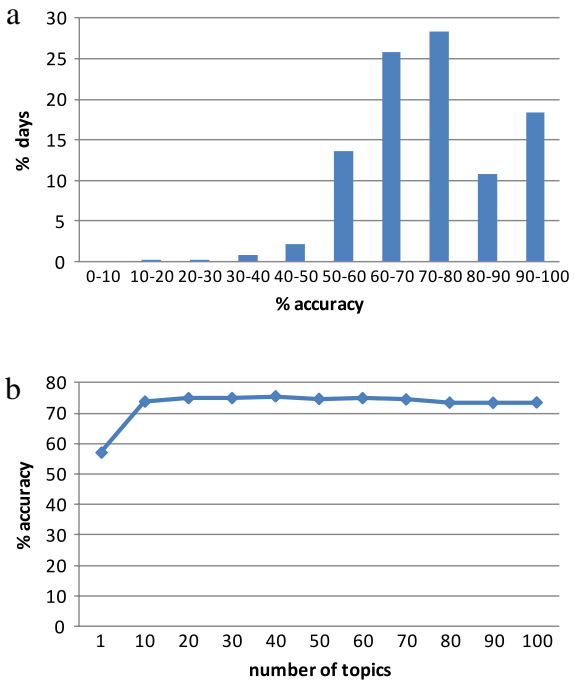


Fig. 6. (a) Average day reconstruction accuracy computed over all users and days. (b) Average day reconstruction accuracy (y-axis) as a function of the number of topics extracted (x-axis) from the user's dataset.

6. LDA-topic prediction

6.1. Method

Given a day of the user with some whereabouts information missing, we want to predict where the user has been or will be at that time. Our approach is based on the following steps:

1. Given a dataset of past user whereabouts we compute N LDA-topics representing user's routine behavior.
2. For each topic z, we compute the probability that the day with missing information expresses the routine encoded in z with 2 different approaches. If the system has to predict all the future *time-periods* of the user in that day, we compute $p(d|z) = \prod_k p(w_k|z)$ where k are all the known words composing the unseen document. If the system has to predict a single missing *time-period*, we use in the computation of $p(d|z)$ only the words that are in the *time-periods* before and after the missing one. The idea is that the closer the words are to the *time-period* we removed, the more important they are in $p(d|z)$ computation. This latter prediction task is valuable in reconstructing missing information in the dataset and directly compares to related work [6].
3. For each missing location word i, we have to select the most probable label $L_i \in \{H, W, E, N\}$ where the user might have been. For each label L_i we compute $p(L_i)$ according the following procedure:
   (a) For each topic z, we define d_z as the document maximizing $p(d|z)$. This is the most representative day for the topic z.
   (b) We define the set $Z(L_i)$ as the set of topics in which d_z has the label L at the time slot i. Generally speaking this is the set of topics that would predict that the missing location word i is L.
   (c) We compute $p(L_i) = \sum_{z \in Z(L_i)} p(d|z) \cdot p(d_z|z)$ where d is the current document we are trying to predict missing labels and $p(d|z)$ is computed as in point 2.
4. Finally we predict L_i as the label with the maximum $p(L_i)$.

This allows to predict the routines (topics) the user will undergo in the future given the routine performed in the previous part of the day, and to predict the user's whereabouts on the basis of the predicted routine. It is worth noticing that in contrast with the work presented in [6], our mechanism takes into consideration all the topics.

6.2. Experiments and discussion

The experiment setup to test our prediction method has been conducted on the Reality Mining data. We first present experiments in which we try to predict a single-missing time period. Then we present experiments trying to predict all future time periods.

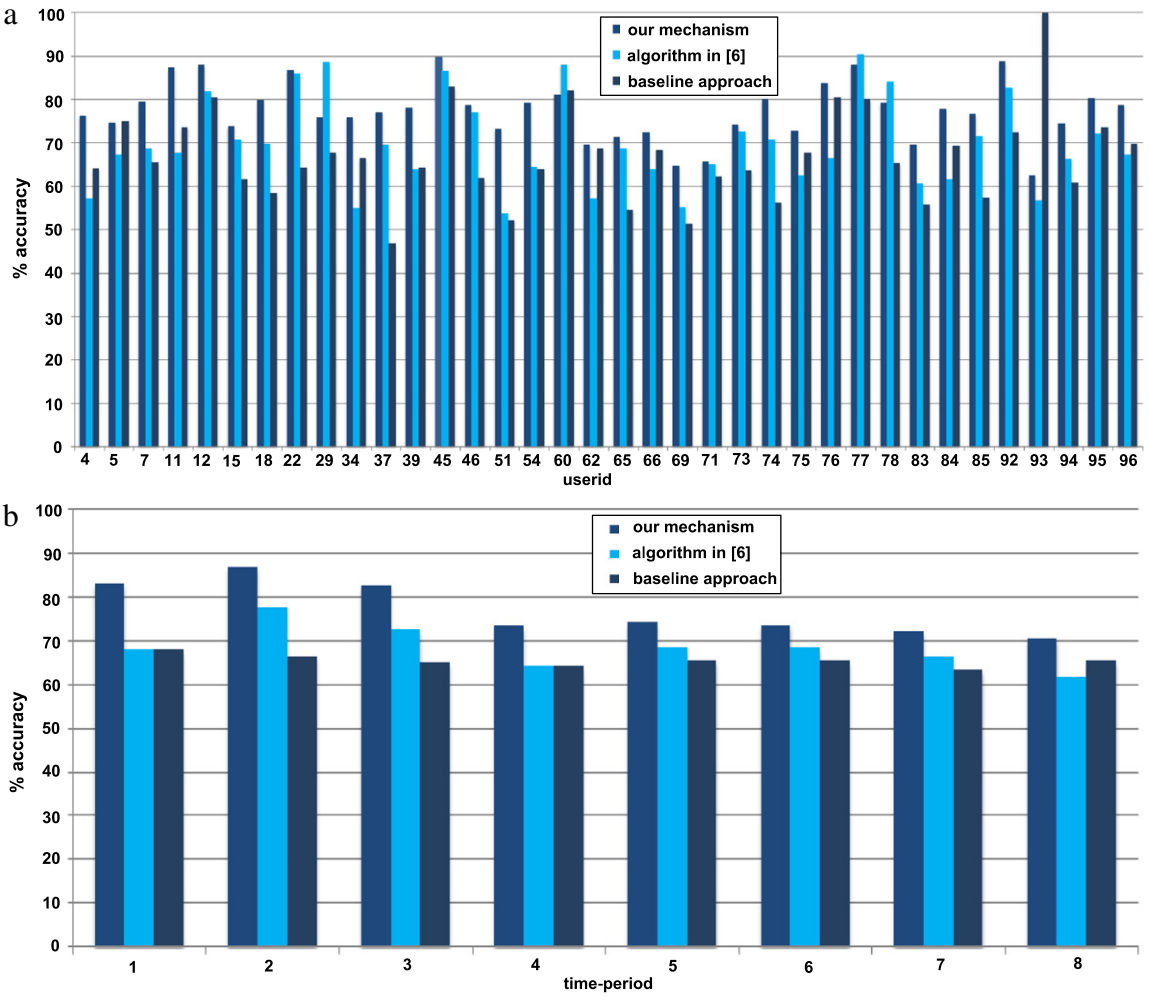


Fig. 7. (a) Average prediction accuracy as a function of users. (b) Average prediction accuracy as a function of *time-periods*.

A first experiment is based on the following key points:

1. We consider the set of all the days of all the users.
2. Following a leave-one-out approach, for each day d, we constructed a training set consisting of all the days other than d, and we extracted 100 LDA-topics from it.
3. d is the testing day whose words will be predicted. In particular, we remove words of a given *time-period* and we evaluate the prediction accuracy in reconstructing the day.
4. Classification accuracy is obtained as an average accuracy over all days, removed time periods, and users taken into consideration.

Fig. 7 shows the location accuracy computed over users and *time-periods*. Fig. 7.a shows prediction accuracy as a function of the users taken into consideration. Results are compared with both the algorithm proposed in [6] and a baseline approach. The algorithm proposed in [6] replaces missing labels with those of the top day for the most likely topic associated to the day with missing labels.[3] The baseline approach predicts a label using the most likely label coming from the same time and day of all the other weeks. Our mechanism produces better performances on average 8% better than the other algorithms. Fig. 7.b shows prediction accuracy as a function of the *time-periods* to be predicted. It is worth noting that higher accuracy is obtained in predicting early morning *time-periods* (1, 2, 3) than rest-of-the-day *time-periods* (4, 5, 6, 7, 8). This is because user behavior is intrinsically less variable at early morning (the user is often at home or has the cell phone switched off) (see for comparisons [24]). Our mechanism produces better performances in all the *time-periods* on average 10% better than all the other algorithms.

[3] It is worth noting the accuracy presented in Fig. 7 is higher than the performance presented in [6] – about 40%–50% – since we considered a different feature vector.

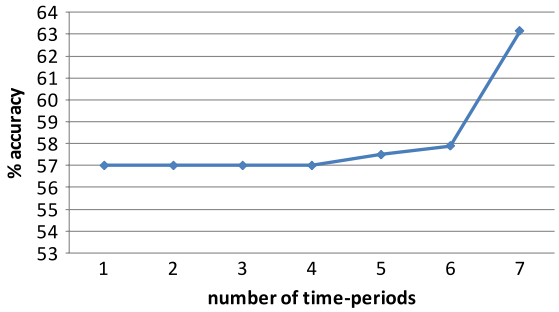
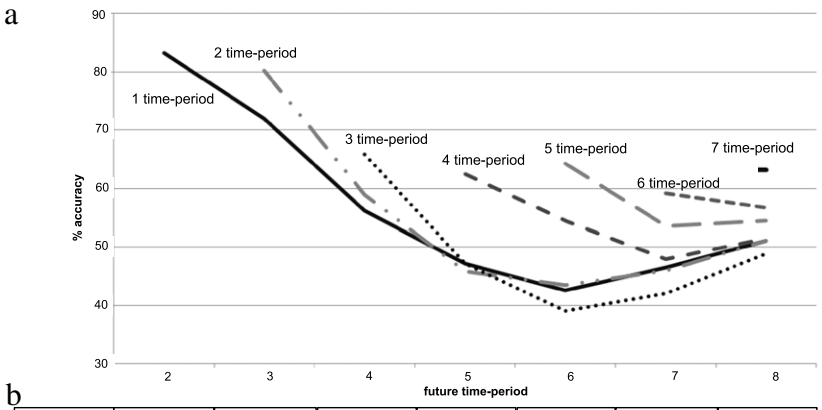


Fig. 8. Average prediction accuracy (*y*-axis) obtained over all users and days as a function of the number of subsequent *time-periods* (*x*-axis) used as the basis to predict the remaining part of the day.



TIME-PERIOD	2	3	4	5	6	7	8
% ENTROPY	47,50	51,80	53,77	52,05	52,29	53,29	37,16

Fig. 9. (a) Average prediction accuracy (*y*-axis) obtained over all users and days as a function of the subsequent time slots (*x*-axis). The reported performance is obtained using a different number of increasing *time-periods* as portion of the day already known. (b) The figure also reports entropy percentage associated to each *time-period*.

In a second experiment we use the same former setup but, instead of just removing a single *time-period*, we test our mechanism in predicting users' subsequent whereabouts using only the information from a portion of the day. Fig. 8 shows the average prediction accuracy obtained by using an increasing number of observed *time-periods*. For the sake of comparison it is possible to see that accuracy is lower than in Fig. 7. This is because of the added complexity in predicting multiple future *time-periods*.

To better analyze the prediction accuracy in this experiment, Fig. 9.a shows, using an increasing number of *time-periods*, the accuracy for each individual subsequent period. For example, given information on where the user has been in the 1st *time-period*, we measured prediction accuracy for the 2nd, 3rd, …, 8th. Fig. 9.b also reports information entropy associated to each *time-period*. The higher the entropy, the more intrinsically difficult is the prediction task. Looking at the graph it is possible to see a general decreasing trend in accuracies the more the algorithm try to predict in the future. However, accuracy in the central part of the day drops because the high entropy in user whereabouts at that time. This phenomenon is particularly evident in the plot associated to the knowledge of the first 3 time periods. Prediction accuracy drops in the central part of the day (time periods 6 and 7) because the high entropy confuses the classifier (see for comparisons [24]).

7. Conclusion and future work

In this paper we presented a methodology to automatically classify the routine whereabouts extracted from a mobility dataset with meaningful labels. We also presented a novel prediction mechanism to infer user's future whereabouts. These two mechanisms can be combined to describe from an high-level perspective the current and future whereabouts of the user.

Our future work in this area will target 3 main directions: (i) We will apply the presented approaches to "live" datasets such as those that can be acquired from Google Latitude [26] and Twitter [27]. (ii) We will develop mechanisms to add/modify topic labels at run time, so as to enable the use of the system in a wide range of scenarios. (iii) We will develop Web-based visualization mechanisms to inspect and communicate whereabouts behaviors in an effective way (http://unfoldingmaps.org).

The ultimate goal will be to create a real live Web application allowing different classes of users to see, understand and predict their own and other users' whereabouts.

Acknowledgments

The work was supported by the SAPERE (Self-Aware Pervasive Service Ecosystems) project (EU FP7-FET, Contract No. 256873). We also thank Dr. Katayoun Farrahi for clarifications on their use of the LDA algorithm on the Reality Mining dataset.

References

[1] S. Patel, J. Kientz, G. Hayes, S. Bhat, G. Abowd, Farther than you may think: an empirical investigation of the proximity of users to their mobile phones, in: International Conference on Ubiquitous Computing, Orange County (CA), USA, 2006.
[2] T. Choudhury, S. Consolvo, B. Harrison, J. Hightower, A. LaMarca, L. LeGrand, A. Rahimi, A. Rea, G. Bordello, B. Hemingway, P. Klasnja, K. Koscher, J. Landay, J. Lester, D. Wyatt, D. Haehnel, The mobile sensing platform: an embedded activity recognition system, IEEE Pervasive Computing 7 (2) (2008) 32–41.
[3] M. Massimi, K. Truong, D. Dearman, G. Hayes, Understanding recording technologies in everyday life, IEEE Pervasive Computing 9 (3) (2010) 64–71.
[4] T. Strang, C. Linnhoff-Popien, A context modeling survey, in: Workshop on Advanced Context Modelling, Reasoning and Management, Nottingham, UK, 2004.
[5] N. Eagle, A. Pentland, Eigenbehaviors: identifying structure in routine, Behavioral Ecology and Sociobiology 63 (7) (2009) 1057–1066.
[6] K. Farrahi, D. Gatica-Perez, Probabilistic mining of socio-geographic routines from mobile phone data, IEEE Journal of Special Topics in Signal Processing 4 (4) (2010) 746–755.
[7] D. Blei, A. Ng, M. Jordan, Latent dirichlet allocation, Journal of Machine Learning Research 3 (1) (2003) 993–1022.
[8] J. Gemmell, G. Bell, R. Lueder, Mylifebits: a personal database for everything, Communications of the ACM 49 (1) (2006) 88–95.
[9] A. Williams, P. Dourish, Reimagining the city: the cultural dimensions of urban computing, IEEE Computer 39 (9) (2006) 38–43.
[10] R. Mayrhofer, H. Radi, A. Ferscha, Recognizing and predicting context by learning from user behavior, special issue on mobile multimedia, Radiomatics: Journal of Communication Engineering 1 (1) (2004) 30–42.
[11] A. Schmidt, P. Holleis, J. Hakkila, E. Rukzio, R. Atterer, Mobile phones as tool to increase communication and location awareness of users, in: Mobility, Bangkok, Thailand, 2006.
[12] C. Bishop, Pattern Recognition and Machine Learning, Springer Verlag, 2006.
[13] F. Calabrese, J. Reades, C. Ratti, Eigenplaces: analysing cities using the space–time structure of the mobile phone network, IEEE Pervasive Computing 9 (1) (2010) 78–84.
[14] Q. Mei, X. Shen, C. Zhai, Automatic labeling of multinomial topic models, in: ACM International Conference on Knowledge Discovery and Data Mining, San Jose (CA), USA, 2007.
[15] J. Kang, W. Welbourne, B. Stewart, G. Borriello, Extracting places from traces of locations, in: ACM International Workshop on Wireless Mobile Applications and Services on WLAN Hotspots, Philadelphia (PA), USA, 2004.
[16] N. Bicocchi, G. Castelli, M. Mamei, A. Rosi, F. Zambonelli, Supporting location-aware services for mobile users with the whereabouts diary, in: International Conference on MOBILe Wireless MiddleWARE, Operating Systems, and Applications, Innsbruck, Austria, 2008.
[17] Y. Zheng, L. Zhang, X. Xie, W.-Y. Ma, Mining interesting locations and travel sequences from gps trajectories for mobile users, in: International World Wide Web Conference, Madrid, Spain, 2009.
[18] J. Hightower, S. Consolvo, A. LaMarca, I. Smith, J. Hughes, Learning and recognizing the places we go, in: International Conference on Ubiquitous Computing, Tokyo, Japan, 2005.
[19] D. Kim, J. Hightower, R. Govindan, D. Estrin, Discovering semantically meaningful places from pervasive rf-beacons, in: International Conference on Ubiquitous Computing, Orlando (FL), USA, 2009.
[20] T. Sohn, A. Varshavsky, A. LaMarca, M. Chen, T. Choudhury, I. Smith, S. Consolvo, J. Hightower, W. Griswold, E. de Lara, Mobility detection using everyday gsm traces, in: International Conference on Ubiquitous Computing, Orange County (CA), USA, 2006.
[21] J. Froehlich, J. Krumm, Route prediction from trip observations, in: Intelligent Vehicle Initiative, Technology Advanced Controls and Navigation Systems, SAE World Congress and Exhibition, Detriot (MI), USA, 2008.
[22] F. Giannotti, M. Nanni, D. Pedreschi, F. Pinelli, Trajectory pattern mining, in: ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Jose (CA), USA, 2007.
[23] V.V.M. Sricharan, A pragmatic analysis of user mobility patterns in macrocellular wireless networks, Pervasive Mobile Computing Journal 4 (5) (2008) 616–632.
[24] D. Choujaa, N. Dulay, Predicting human behaviour from selected mobile phone data points, in: International Conference on Ubiquitous Computing, Copenhagen, Denmark, 2010.
[25] K. Farrahi, D. Gatica-Perez, Daily routine classification from mobile phone data, in: Workshop on Machine Learning and Multimodal Interaction, Utrechts, Netherlands, 2008.
[26] L. Ferrari, M. Mamei, Discovering daily routines from google latitude with topic models, IEEE Workshop on Context Modeling and Reasoning, Seattle (WA), USA, 2011.
[27] L. Ferrari, A. Rosi, M. Mamei, F. Zambonelli, Extracting urban patterns from location-based social networks, in: ACM SIGSPATIAL International Workshop on Location-based Social Networks, Chicago (IL), USA, 2011.