ARTICLE IN PRESS

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

Contents lists available at SciVerse ScienceDirect

Pervasive and Mobile Computing

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

Where to go from here? Mobility prediction from instantaneous information

Vincent Etter*, Mohamed Kafsi, Ehsan Kazemi, Matthias Grossglauser, Patrick Thiran

School of Computer and Communication Sciences, EPFL, CH-1015 Lausanne, Switzerland

ARTICLE INFO

Article history:
Available online xxxx

Keywords:
Machine learning
Mobility prediction
Probabilistic Graphical Models
Dynamical Bayesian Network
Artificial Neural Networks
Gradient Boosted Decision Trees

ABSTRACT

We present the work that allowed us to win the Next-Place Prediction task of the Nokia Mobile Data Challenge. Using data collected from the smartphones of 80 users, we explore the characteristics of their mobility traces. We then develop three families of predictors, including tailored models and generic algorithms, to predict, based on instantaneous information only, the next place a user will visit. These predictors are enhanced with aging techniques that allow them to adapt quickly to the users' changes of habit. Finally, we devise various strategies to blend predictors together and take advantage of their diversity, leading to relative improvements of up to 4%.

1. Introduction

Mobility is a central aspect of our life; the locations we visit reflect our tastes and lifestyle and shape our social relationships. The ability to predict the places a user will visit is therefore beneficial to numerous applications, ranging from forecasting the dynamics of crowds to improving the relevance of location-based recommendations.

This paper reports our winning contribution to the Nokia Mobile Data Challenge (NMDC). Our task was [1] "to predict the next destination of a user given the current context, by building user-specific models that learn from their mobility history, and then applying these models to the current context to predict where the users go next". A context is described by the data collected from the mobile phone of the user (date, location of the user, cell tower id, WLAN, phone calls, etc.).

First, we examined carefully the data and their characteristics and implemented techniques to overcome some of the roots of unpredictability. For instance, we noticed that some users change their home location during the observation period so we developed aging techniques that allow us to detect and adapt to these changes. Then, we developed several mobility predictors, based on graphical models, neural networks, and decision trees. These predictors exhibited close average prediction accuracies, yet we observed, for each user, a high performance variability between predictors.

In order to take advantage of this variability, we finally combined these predictors using different blending strategies. Blending is an ensemble method that combines different predictors in order to obtain a predictor which is more accurate than any of the individual ones. We submitted five sets of predictions to the challenge, based on these blending techniques. Each submission obtained a higher prediction accuracy than all the submissions from other participants, allowing our team to win the first place of the challenge.

The paper is organized as follows: we describe the NMDC and its dataset in Section 2. In Section 3, we introduce a framework where we define the notations, the learning process and the prediction performance measure. Then, we present,

* Corresponding author. Tel.: +41 216931260.
E-mail addresses: vincent.etter@epfl.ch (V. Etter), mohamed.kafsi@epfl.ch (M. Kafsi), ehsan.kazemi@epfl.ch (E. Kazemi), matthias.grossglauser@epfl.ch (M. Grossglauser), patrick.thiran@epfl.ch (P. Thiran).

1574-1192/$ – see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.pmcj.2013.07.006

in Section 4, various models serving as basis for our predictors, and we show their individual performance. In Section 5, we describe different blending strategies and establish their performance gain over individual predictors. Finally in Section 6, we briefly discuss other works related to predicting the mobility of users, before concluding in Section 7.

2. Nokia mobile data challenge: next-place prediction

The NMDC is [1] "a large-scale research initiative aimed at generating innovations around smartphone-based research, as well as community-based evaluation of related mobile data analysis methodologies". It was organized by Nokia and took place from January 2012 to April 2012. It featured an open track, in which participants were able to propose their own problem to study, and three dedicated tracks, each defining a specific problem for teams to solve: semantic place prediction, next-place prediction and demographic attributes' prediction.

At the heart of this challenge is the dataset gathered during the Lausanne Data Collection Campaign (LDCC) [2]. This dataset consists of a rich set of features (locations, phone calls, text messages, application usage, etc.) recorded from the smartphones of 170 participants, over periods of time ranging from a few weeks to almost two years. These data were collected in a privacy-preserving manner, allowing for meaningful statistics to be gathered while the anonymity of participants was protected.

Each task had its own subset of the LDCC data. The Next-Place Prediction task, the focus in this paper, was assigned a subset of 80 users. For each user, the last 50 days of data were kept as a test set for the evaluation of each team's submissions, and the rest was used as training data.

For privacy reasons, all identifiers (phone numbers, WLAN SSIDs, contact names, etc.) were encrypted, but more importantly, physical locations were not released. Instead, for each user, the organizers of the NMDC first identified *places* – corresponding to discs with a 100 m radius – by using both GPS and WLAN data. Then, they represented each place by a unique identifier. Consequently, a sequence of geographic coordinates is represented as a sequence of place identifiers.

These visits represent the basic unit for the prediction task. They are defined by their starting and ending times, and the corresponding place. In addition, several types of data are available: accelerometer, application usage, GSM, WLAN, media plays, etc. The complete list can be found in the dataset description [1]. Given a visit and the data characterizing it, our task was to predict the next place visited by the user.

At the end of the challenge, each participating team was allowed to submit five different sets of predictions, corresponding to visits from the undisclosed part of the LDCC data. Then, the organizers of NMDC evaluated the prediction accuracy of each participating team's submissions.

We present below two major constraints (imposed by the rules of the NMDC) that restrict the range of methods we can use and make our task more challenging.

User specificity. To prevent cross-referencing people and places between users, all sensitive data are user-specific: the identifiers are encrypted using different keys, and places are defined and numbered for each user independently. Moreover, the rules of the challenge explicitly forbade all participants to try and reverse this process, or make some links between users. We were therefore not allowed to build joint models over the user population, *i.e.*, to learn from one user to make a prediction about another. For this reason, we built user-specific predictors and consider each user independently.

Memoryless predictors. As explained above, the input for the Next-Place Prediction task is the *current* visit, along with all additional data recorded from the user's phone during that time. However, we do not have access to the *history* of the user, *i.e.*, the sequence of previous visits. If we did, we could develop higher-order predictors that not only take into account the current place but also the sequence of places visited just before. Indeed, such information is very useful: if a user is currently at a transportation hub, *e.g.* a bus station, knowing whether he was home or at work just before greatly helps in predicting his next move. Because this information is not available to us in this challenge, we limit ourselves to *memoryless* predictors, *i.e.*, methods that take into account only the current context, without any knowledge of the past.

3. Place prediction framework

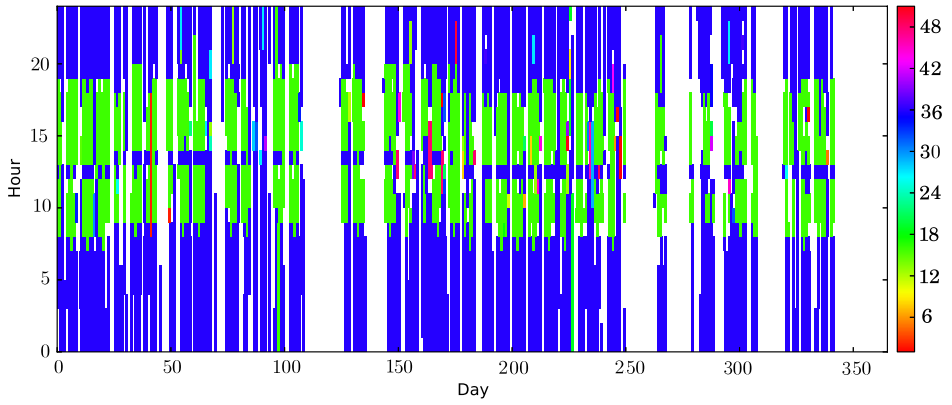
In this section, we explore some characteristics of the dataset and define the framework within which we develop our predictors.

3.1. Dataset characteristics

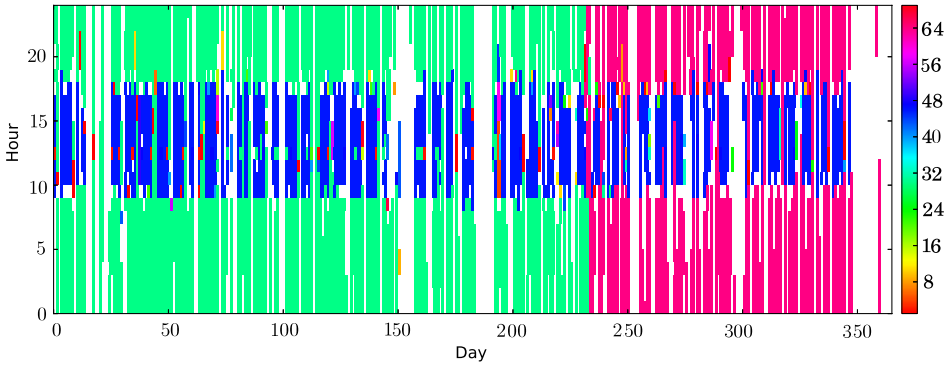
We show in Fig. 1 an intuitive representation of the mobility traces of three users selected from the dataset. The figure depicts a user's behavior over a year as a matrix, where each column is a day of the year and each line an interval of 1 h. We map each place to a color and leave blank intervals of time during which we have no information about the user's location.

User 143, whose mobility is represented in Fig. 1(a), has a very regular behavior, which seems to support the results (such as those presented by Song et al. [3]) claiming that human mobility is very predictable. However, similarly to User 1

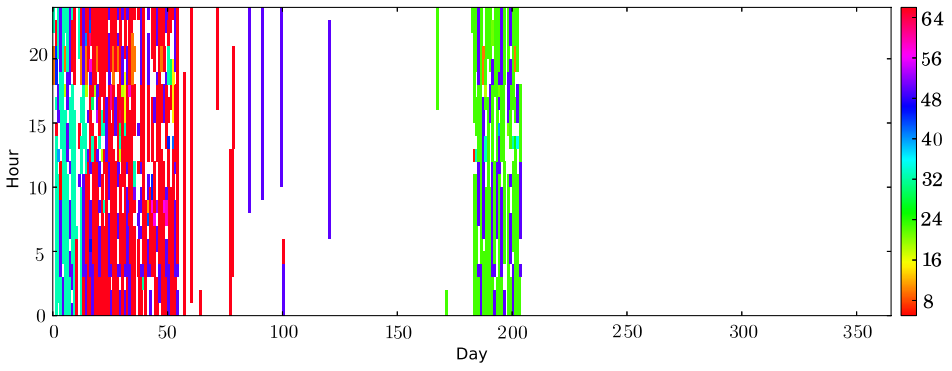
Please cite this article in press as: V. Etter, et al., Where to go from here? Mobility prediction from instantaneous information, Pervasive and Mobile Computing (2013), http://dx.doi.org/10.1016/j.pmcj.2013.07.006



(a) User 143: regular mobility pattern.



(b) User 13: regular mobility pattern with change of home.



(c) User 1: irregular mobility pattern characterized by data gaps and non-stationarity.

Fig. 1. Behavior of users over a year, shown as matrices where each column is a day of the year and each line an interval of 1 h. We map each place to a color and leave blank intervals of time during which we have no information about the user's location. (a) illustrates the behavior of a very regular user, (b) a home change, and (c) data gaps and non-stationarities.

(Fig. 1(c)), the majority of users show no clear regular pattern in their behavior. Of course, a lack of visual regularity does not imply that there is no underlying structure in a user's mobility. We will see in Section 4.4 that we can still predict the behavior of such users with a reasonable accuracy.

We summarize below some salient characteristics of the data that we believe are critical to the prediction task.

Non-stationarity. We often observe a significant change in users' habits over time, as illustrated in Fig. 1(b). The fact that some users change their home or work location right at the end of the observation period complicates the prediction task. To overcome this, we implement aging mechanisms, as described in Section 4.1.1. Moreover, to get a realistic estimation of our predictors' performances, we keep the last part of the dataset as testing data, as explained in Section 3.3.

Please cite this article in press as: V. Etter, et al., Where to go from here? Mobility prediction from instantaneous information, Pervasive and Mobile Computing (2013), http://dx.doi.org/10.1016/j.pmcj.2013.07.006

Table 1
List of the definition and domain of the variables relative to a user, as well as those describing his nth visit.

Variable	Domain	Explanation
L	$\mathbb{N}$	Number of distinct places
$\mathcal{L}$	$\{1, \ldots, L\}$	Set of visited places
k	$\mathbb{N}$	Time resolution
$X(n)$	$\mathcal{L}$	Place
$T_s(n)$	$\mathbb{N}$	Absolute starting time
$H_s^k(n)$	$\{1, \ldots, k\}$	Quantized starting hour
$D_s(n) = \text{day}(T_s(n))$	$\{1, \ldots, 7\}$	Starting day
$W_s(n) = \text{weekday}(T_s(n))$	$\{0, 1\}$	Indicates whether the visit starts on a weekday
$T_e(n)$	$\mathbb{N}$	Absolute ending time
$H_e^k(n)$	$\{1, \ldots, k\}$	Quantized ending hour
$D_e(n) = \text{day}(T_e(n))$	$\{1, \ldots, 7\}$	Ending day
$W_e(n) = \text{weekday}(T_e(n))$	$\{0, 1\}$	Indicates whether the visit ends on a weekday
$U(n)$	$\{0, 1\}$	Indicates whether there might be an unobserved place between $X(n)$ and $X(n+1)$
$C(n)$	$\{0, 1\}$	Indicates whether the user charged his phone during the visit

Data gaps. We experience, for some users, periods (ranging from a few hours up to a few months) with no information about their behavior. Moreover, as shown in Fig. 1(c), these gaps are sometimes followed by a change of mobility habits. To limit the effect of such transitions, our predictors take into account the possibility that we have missed some data between two detected visits (Section 3.2).

Sparsity. The period of observation for some users is too short (less than 15 days) to reflect faithfully their mobility patterns. We overcome this lack of data by allowing for coarser segmentations of the day, using the time resolution parameter described in Section 3.2. We also limit the complexity of our predictors, so that they do not over-fit the data.

We believe that taking the above observations into account in the design of predictors has a significant effect on their prediction accuracy.

3.2. Notations

Before formally introducing our predictors, we need to define the variables that describe the dataset. During the study period, a user makes N visits of variable duration to L distinct places, represented by the set $\mathcal{L} = \{1, \ldots, L\}$.

In Table 1, we list the variables corresponding to a user, as well as those relative to his nth visit. All time-relative variables are derived from the starting and ending times, which are given as absolute times. The binary variable $U(n)$ indicates whether there might be an unobserved place between $X(n)$ and $X(n+1)$. This situation typically arises when location data are partly available between the two visits. In such a case, we say that the transition from $X(n)$ to $X(n+1)$ is not necessarily *direct*. The directness of a transition is given as a feature in the NMDC dataset.

To allow for various quantizations of the day, we introduce a *time resolution* parameter k. This lets us consider a coarser segmentation of the day: instead of always splitting a day into 24 h, we can choose to split it into k time periods. For instance, if $k = 2$, $H_s^k(n) \in \{1, 2\}$, with $H_s^k(n) = 1$ corresponding to the nth visit starting between midnight and noon. Such a coarse segmentation can be helpful when training predictors for a user for which few data are available.

3.3. Learning procedure

For each user, we separate the data into three parts, as illustrated in Fig. 2: we define the first 80% of the data as set A, the following 10% as set B and the last 10% as set C. Finally, we call set D the undisclosed part of the data, on which our predictors were evaluated during the final part of the NMDC.

The reason we divide the dataset deterministically is based on the non-stationarity of the users' behavior, as described in Section 3.1. In fact, we expect set D to be much more similar to the end of the dataset than to its beginning. Indeed, even if a user's behavior is globally non-stationary, it usually shows regular patterns over smaller time intervals. Having set C as close as possible to set D maximizes the likelihood of their samples belonging to the same "stationary" period. Moreover, by training our predictors on "past" data and evaluating them on very recent data, we can test whether they are able to adapt to users' changes of habit.

For each predictor, the training is performed in three parts: first, we train on set A and evaluate the performance on set B, to compare individual predictors. Then, we train on both sets A and B and evaluate the prediction accuracy on set C, with different blending strategies. Finally, we train on sets A, B and C in order to predict for the samples in set D.

3.4. Performance measure

To evaluate the performance of a predictor on a set of visits, we consider its prediction accuracy, *i.e.*, the proportion of samples for which it successfully predicts the next place.

Please cite this article in press as: V. Etter, et al., Where to go from here? Mobility prediction from instantaneous information, Pervasive and Mobile Computing (2013), http://dx.doi.org/10.1016/j.pmcj.2013.07.006

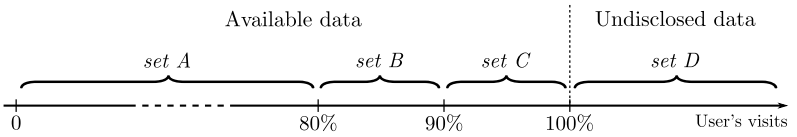


Fig. 2. Separation of a user's dataset. We define the first 80% of the user's visits as set *A*, the following 10% as set *B*, and the last 10% as set *C*. Finally, we call set *D* the undisclosed part of the dataset, on which the final performances are computed.

First, consider a predictor ϕ: it takes as input $\boldsymbol{v}^{(n)}$, the data corresponding to the nth visit, and outputs a probability distribution over the possible next places. More formally, a predictor is a function

$$\phi: \mathcal{V} \to \left\{ \boldsymbol{x} \in [0,1]^L : \sum_{l=1}^{L} x_l = 1 \right\},$$

where $\mathcal{V}$ is the space of data corresponding to a visit. We could directly define the output of a predictor as the predicted next place. However, keeping a distribution over places as output allows us to combine predictors. Indeed, we can easily put several predictors together by computing a mixture of their outputted probability distributions over places. As explained in Section 5, there are different ways of choosing the weight of each predictor, each resulting in a unique global predictor.

The place $\hat{X}_n^{\phi}$ predicted by ϕ for the visit $\boldsymbol{v}^{(n)}$ is thus the most likely next place

$$\hat{X}_n^{\phi} = \arg\max_{l \in \mathcal{L}} (\phi(\boldsymbol{v}^{(n)}))_l, \tag{1}$$

where $(\phi(\boldsymbol{v}^{(n)}))_l$ is the lth component of the vector outputted by ϕ when given the data corresponding to the nth visit as input, *i.e.*, the probability that the next visited place is l.

Finally, we define the prediction accuracy $A_S(\phi)$ of the predictor ϕ over the samples in set S as

$$A_S(\phi) = \frac{1}{|S|} \sum_{i \in S} \mathbb{I}_{\left\{\hat{X}_i^{\phi} = X(i+1)\right\}}, \tag{2}$$

where $|S|$ is the number of samples in set S, $X(i+1)$ is the true next place corresponding to the ith visit, and $\mathbb{I}_{\{A\}}$ is the indicator function, taking value 1 if the event A is true, and 0 otherwise.

4. Predictors

In this section, we present the techniques we use to build predictors. First, we introduce a Dynamical Bayesian Network (DBN), which we tailor specifically for this challenge and its data characteristics. A DBN involves a modeling phase where we express causal relationships and independence assumptions between the features of the visits. Then, we present more generic methods that require no specific assumption about the input variables. The generic methods we implement are Artificial Neural Networks (ANN) and Gradient Boosted Decision Trees (GBDTs). We use both a crafted model and non-linear methods, because the former offers an intuitive representation of the problem and could benefit from our insights, while the latter have shown to give good performances in many similar prediction problems. Finally, we summarize the performances of the predictors in Section 4.4.

4.1. Dynamical Bayesian Network (DBN)

We model the mobility patterns of individuals as a DBN. The rationale behind our model is as follows: the next place a user will visit depends on his current place and on the time at which he leaves it. The dependence between the current and next place is strong when the difference between the ending time of the current visit and the starting time of the next one is small (typically the case for direct transitions). However, as this time difference gets larger, the influence of the present place on the next one fades out while the starting time of the next visit bears increasing importance.

As we do not know the starting time of the next visit, the main challenge is to model its randomness, given the carefully chosen information about the current visit. As shown in Fig. 3, our DBN captures these intuitions: the conditional distribution of the next place $p\left(X(n+1)|X(n), H_e(n), U(n), W_e(n)\right)$ is a linear combination of place- and time-dependent distributions

$$\pi p\left(X(n+1)|X(n)\right) + (1-\pi) p\left(X(n+1)|H_e(n), W_e(n), U(n)\right),$$

where $0 \leq \pi \leq 1$ is the parameter that governs the contribution of each distribution. For ease of notation, we omit the time resolution parameter k and assume that it is fixed. The place-dependent component

$$p(X(n+1)|X(n)) \tag{3}$$

is simply a first-order Markov chain that encodes the frequency of transitions between places. Using Bayes' rule, we express the time-dependent distribution

$$p\left(X(n+1)|H_e(n), W_e(n), U(n)\right) \tag{4}$$

Please cite this article in press as: V. Etter, et al., Where to go from here? Mobility prediction from instantaneous information, Pervasive and Mobile Computing (2013), http://dx.doi.org/10.1016/j.pmcj.2013.07.006

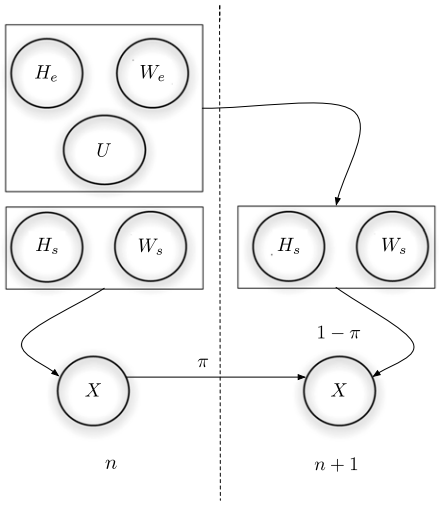


Fig. 3. Diagram of a DBN where nodes represent random variables and edges probabilistic dependencies between them. The conditional distribution of the next place $p\,(X(n+1)|X(n), H_e(n), U(n), W_e(n))$ is a linear combination of place-dependent $p\,(X(n+1)|X(n))$ and time-dependent $p\,(X(n+1)|H_e(n), W_e(n), U(n))$ distributions. Note that the structure of the DBN reflects the conditional independence of $X(n+1)$ and $(H_e(n), W_e(n), U(n))$ given $(H_s(n+1), W_s(n+1))$.

as

$$\sum_{W_s}\sum_{H_s}\big\{p(X(n+1)|H_s(n+1), W_s(n+1), H_e(n), W_e(n), U(n))p(H_s(n+1), W_s(n+1)|H_e(n), W_e(n), U(n))\big\}.$$

Note that the conditional distribution

$$p\,(H_s(n+1), W_s(n+1)|H_e(n), W_e(n), U(n)) \tag{5}$$

models the randomness of the starting time of the next visit $(H_s(n+1), W_s(n+1))$ given the ending time of the current one $(H_e(n), W_e(n))$ and the directness of the transition $U(n)$. In addition to reflecting the temporal rhythm at which a specific user moves from one place to another, the conditional distribution (5) also captures the randomness of the data gaps. Empirically, we observe that direct transitions usually imply a shorter interval of time between the visits. This is not surprising: if the transition between the nth and $(n+1)$th visits is direct $(U(n) = 0)$, then we are sure that there are no intermediate visits between them. The main assumption we make when designing our DBN is that $X(n+1)$ is independent of $H_e(n)$, $W_e(n)$ and $U(n)$ given $H_s(n+1)$ and $W_s(n)$. We can therefore write (4) as

$$\sum_{W_s}\sum_{H_s}\big\{p\,(X(n+1)|H_s(n+1), W_s(n+1))\,p\,(H_s(n+1), W_s(n+1)|H_e(n), W_e(n), U(n))\big\}.$$

The assumption of independence makes sense, as knowing the time $(H_e(n), W_e(n))$ at which a user leaves his current place is not informative (with respect to the next place $X(n+1)$) if we know the starting time of the next visit $(H_s(n+1), W_s(n+1))$.

The choice of the model structure and variables is driven by our intuition and confirmed by empirical evidence. We tested several variants of our model: for example, we incorporated in our DBN the distribution $p(X(n+1)|X(n), U(n))$ instead of the distribution $p(X(n+1)|X(n))$ to check whether the directness of the transition contains information about the next place. However, the prediction accuracy decreased. Furthermore, the lack of data prohibits us from learning more sophisticated distributions since over-fitting a small training set leads to very poor generalization.

To predict the user's next place using our model, we estimate the DBN parameters π, $p(X(n+1)|X(n))$, $p\,(X(n+1)|H_s(n+1), W_s(n+1))$ and $p\,(H_s(n+1), W_s(n+1)|H_e(n), W_e(n), U(n))$. We take two different approaches.

- We learn the distributions by counting the frequency of given realizations and then choosing the parameter π that maximizes the prediction accuracy on a validation set (subset of set A).
- We formulate the mixture of distributions with respect to a latent variable $\boldsymbol{z}$: we introduce an N-dimensional binary random variable $\boldsymbol{z}$ that indicates, for each visit, the distribution from which it was sampled. In other words, $\boldsymbol{z}_i = 1$ means that the ith visit is sampled from the place-dependent distribution (3), whereas $\boldsymbol{z}_i = 0$ implies that it is sampled from the time-dependent distribution (4). We then use an *Expectation–Maximization* algorithm [4] to maximize the likelihood of the data with respect to the model parameters. Moreover, the structure of the DBN allows us to derive closed form expressions for the update of the model parameters.

Please cite this article in press as: V. Etter, et al., Where to go from here? Mobility prediction from instantaneous information, Pervasive and Mobile Computing (2013), http://dx.doi.org/10.1016/j.pmcj.2013.07.006

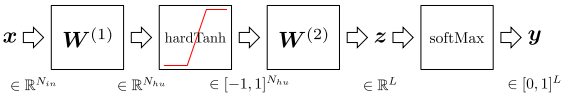


Fig. 4. Architecture of our 2-layer Artificial Neural Network, with N_{in} inputs, N_{hu} hidden units and L outputs. A non-linear transfer function is applied between the first and second layer, and a softmax function is applied to the output, to obtain a probability distribution over places.

4.1.1. Aging to overcome non-stationarity

The idea of introducing *aging* mechanisms in the learning process is based on the observation that, when a user changes his habits, recent history is more representative of his future behavior than the accumulated information.

The first method we use to reduce the negative impact of non-stationarity on the prediction performance is the introduction of an aging mechanism, governed by an aging parameter. The aging parameter, as introduced by Song et al. [5], is a multiplicative factor that intervenes in the learning process to reduce the contribution of old samples. As a result, recent samples will have more impact on the user's mobility model.

The second method is an algorithm that detects changes in home location and adapts the learning process accordingly. At any moment t, we define home as the place where the user spent more than $T_{threshold}$ hours of his sleeping periods during the interval of time $[t - T_{history}, t]$. The parameter $T_{history}$ controls to which extent we keep in memory the past behavior of the user. At the end of the observation period corresponding to the training set, the user who changes his habits will have at least two places flagged as home. We declare the last place flagged as such as his *final home*. More importantly, the history of visits is modified as if the user's home has always been his *final home*. Such modification allows us to capture the user's habits while avoiding the lengthy process of adapting to a home change. The pseudo-code of the home change detection algorithm[1] is shown in Algorithm 1. Empirical results show that applying our home change detection algorithm results in a significant improvement in the prediction accuracy for the users who change their habits during the observation period.

Algorithm 1 Home change detection algorithm

Input: *visits*, $T_{threshold} > 0$, $T_{history} > 0$, *sleeping period*
Output: *visits*
  for all *visits v* **do**
    $t \leftarrow$ starting time of the visit v
    home candidate $\leftarrow$ place where the user spent most of his *sleeping period* in $[t - T_{history}, t]$
    $T_{candidate} \leftarrow$ time spent in *home candidate* during the *sleeping period* in $[t - T_{history}, t]$
    if $(T_{candidate} \geq T_{threshold})$ **then**
      add *home candidate* to *home list*
    end if
  end for
  final home $\leftarrow$ last element of *home list*
  for all *visits v* **do**
    if (*v.place* belongs to *home list*) **then**
      v.place $\leftarrow$ *final home*
    end if
  end for
  return *visits*

4.2. Artificial Neural Networks (ANN)

We can also consider next-place prediction as a classification task: given the current place as input, and potentially some additional features, we define the corresponding class as the next place. With this approach, we train for each user a 2-layer Artificial Neural Network that has N_{in} inputs, N_{hu} hidden units and L outputs. The outputs are normalized to obtain a probability distribution over places. Such a network is illustrated in Fig. 4.

Input encoding. As input, we encode places as categorical data: we represent each place l as a binary vector $\boldsymbol{v} \in \{0, 1\}^L$, where $\boldsymbol{v}_i = 1$ if $l = i$, and 0 otherwise. Other attributes, such as $D_e(n)$ or $H_e^k(n)$, can also be included as additional features and are encoded in a similar way if needed. For example, to use $(X(n), H_s^k(n), D_s(n))$ as inputs, we first encode them as binary vectors, as explained above, and then simply concatenate them. The resulting input vector $\boldsymbol{x}$ is thus of size $N_{in} = L + k + 7$.

Training. To obtain a probability distribution $\boldsymbol{y} \in [0, 1]^L$ from the output $\boldsymbol{z} \in \mathbb{R}^L$ of the second layer, we use a softmax transfer function:

[1] Based on empirical evidence, we choose $T_{history} = 14$ days, $T_{threshold} = 18$ h and *sleeping period* to be between 3 and 6 a.m.

Please cite this article in press as: V. Etter, et al., Where to go from here? Mobility prediction from instantaneous information, Pervasive and Mobile Computing (2013), http://dx.doi.org/10.1016/j.pmcj.2013.07.006

$$\boldsymbol{y}_i = \text{softMax}(\boldsymbol{z}_i) = \frac{\exp(\boldsymbol{z}_i)}{\sum_{j=1}^{L} \exp(\boldsymbol{z}_j)}, \quad i \in \{1, \ldots, L\}.$$

A natural loss function to train such a network is the *negative log-likelihood*. For the output $\boldsymbol{y} \in [0, 1]^L$, corresponding to some input $\boldsymbol{x} \in \mathbb{R}^{N_{in}}$ and the ground truth $\boldsymbol{t} \in \{0, 1\}^L$ (where $\boldsymbol{t}_i = 1$ if i is the true next place, and 0 otherwise), we define the loss as

$$L(\boldsymbol{y}, \boldsymbol{t}) = -\sum_{l=1}^{L} \boldsymbol{t}_l \log(\boldsymbol{y}_l).$$

To find the optimal parameters of the ANN, we minimize the above loss function over the training set.

Parameters. For each user, we consider different subsets of the following features, described in Table 1:

- $X(n)$,
- $D_s(n)$, $D_e(n)$,
- $H_s^k(n)$, $H_e^k(n)$, $k \in \{2, 4, 6, 8, 12, 24\}$,
- $W_s(n)$, $W_e(n)$,
- $C(n)$.

By combining the above features in an exhaustive way,[2] we obtain more than 200 ANNs for each user. We also tested more features, but chose not to use them in the end, as they did not improve the overall prediction performance.

Implementation. We implement our ANNs by using Torch 5 [6], a machine learning framework written in Lua. We use a stochastic gradient descent [7] to train each ANN, and we use early stopping [8] as a regularization technique. For all users, we empirically found that $N_{\text{hu}} = 50$ hidden units were sufficient. To speed up the training, we use hardTanh as the non-linear transfer function between the two layers. It is an approximation of the hyperbolic tangent, that is much faster to evaluate.

4.3. Gradient Boosted Decision Trees (GBDTs)

The third type of predictors we build for the NMDC is Gradient Boosted Decision Trees. A GBDT [9,10] can be used for classification tasks; it is accurate, fast and insensitive to noisy and incomplete data. A GBDT is an ensemble of weak binary decision trees, where all trees consist of two to eight nodes and are learned using boosting methods. In the tree structure, each leaf node belongs to one class and there are conditions on the input features in each interior node. Each one of the GBDT's trees is not a good classifier itself; therefore it is called a weak decision tree.

Boosting is a method for combining weak base classifiers in order to build a classifier whose performance is significantly better than the base classifiers' performance. In boosting methods, the base classifiers are trained in sequence. Each of them is trained using a weighted form of the dataset, in which the weighting coefficient of each data sample depends on the performance of the previous classifiers. After a weak learner is built, weights are updated: weights of misclassified (respectively, correctly classified) samples are increased (respectively, decreased). In a GBDT, base classifiers are weak decision trees. The final prediction model is built by adding up contributions of all the individual small trees. Indeed, the model is an ensemble of the trees that becomes more accurate as the number of trees increases.

We train and build our GBDT predictors using Python. In our setting, the set of classes is $\mathcal{L}$, the places visited by the user. The GBDT classifiers are trained using the information available at each time step n, and the next visited place $X(n+1)$ as the target class.

We use two parameters to control the structure of our GBDTs: the number of trees N_{tree} and the minimum number of observations N_{obs} required to create a terminal node in the trees. As input, we tried to use all the provided information to make our predictors more accurate, but we could not extract meaningful patterns from some of the available data such as accelerometer, bluetooth and WLAN traces. Thus, we use three different subsets of the features described in Table 1, with different time resolutions:

- $S_1 = \{D_e(n), H_e^{24}(n), X(n), U(n)\}$,
- $S_2 = \{D_e(n), H_e^{24}(n), H_s^{24}(n), X(n), U(n)\}$,
- $S_3 = \{D_e(n), W_e(n), H_e^{3}(n), H_e^{8}(n), X(n), U(n)\}$.

We show in Table 2 10 GBDT predictors that are obtained by combining the above sets of features with different internal parameters. Fig. 5 illustrates an example of a weak decision tree, obtained by GBDT 1 from user 143.

The frequency of the visits to certain places during weekdays and the time of these visits reveal, with a high probability, the home and work places of users. We found that some users change their home and work places permanently, whereas

[2] The current place $X(n)$ is always used. We then include either $D_s(n)$ and $H_s^k(n)$, or $D_e(n)$ and $H_e^k(n)$, or both, for varying values of k. The other features $W_s(n)$, $W_e(n)$ and $C(n)$ are only used when both starting and ending day/hour are included. We made this choice to reduce the number of combinations and thus the running time of our experiments.

Please cite this article in press as: V. Etter, et al., Where to go from here? Mobility prediction from instantaneous information, Pervasive and Mobile Computing (2013), http://dx.doi.org/10.1016/j.pmcj.2013.07.006

Table 2
Input variables and internal parameters of our GBDT predictors. We use time, place and directness of the transition as input features.

GBDT	Input set	N_{obs}	N_{tree}
1	S_1	2	100
2	S_1	50	5
3	S_1	100	5
4	S_1	500	5
5	S_1	200	10
6	S_2	100	2
7	S_2	50	5
8	S_2	100	5
9	S_3	50	5
10	S_3	100	5

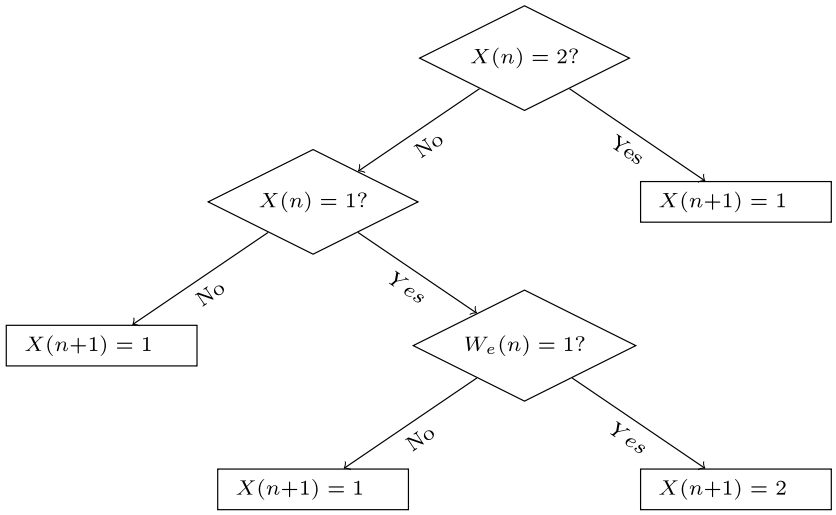


Fig. 5. Example of a weak decision tree, learned by GBDT 1 from user 143. For this user, place 1 is his/her home, and place 2 is his/her workplace.

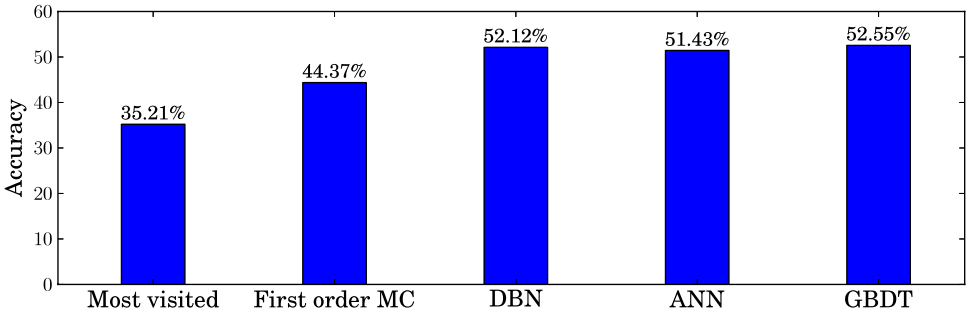


Fig. 6. Accuracy on set *C* of the different families of predictors, trained on sets *A* and *B*, averaged over all users. For each user, we chose the predictor that yields the best prediction accuracy on set *B* when trained on set *A* only. As baselines, we also include one predictor that always outputs the most visited place and one that uses a first-order Markov chain.

others change them temporarily. To use the provided data more efficiently, as they are often sparse, and to make predictors more accurate, we replace the previous home and work places with the new ones and train our GBDT predictors on the modified dataset.

4.4. Results

We select, for each user, the predictor that has the best performance on set *B*, then train it again on both sets *A* and *B*, to evaluate it over set *C*. We show in Fig. 6 the prediction accuracy for each of the three families of predictors presented above, averaged over all users. For comparison, we also include two baseline predictors: the first always predicts the most visited place, while the second uses a first-order Markov chain.

Please cite this article in press as: V. Etter, et al., Where to go from here? Mobility prediction from instantaneous information, Pervasive and Mobile Computing (2013), http://dx.doi.org/10.1016/j.pmcj.2013.07.006

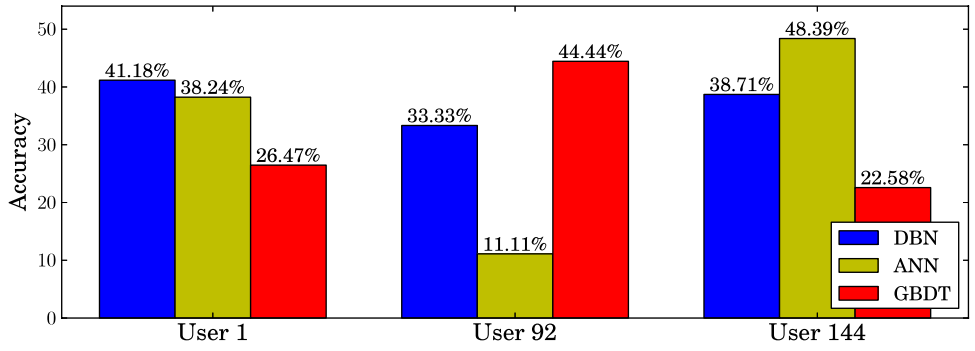


Fig. 7. Accuracy of each family on set *C*, for three selected users. For each family, we chose the best predictor on set *B*. Even though Fig. 6 shows that the three families of predictors have similar average performances, this figure clearly illustrates that, across users, their performances vary greatly.

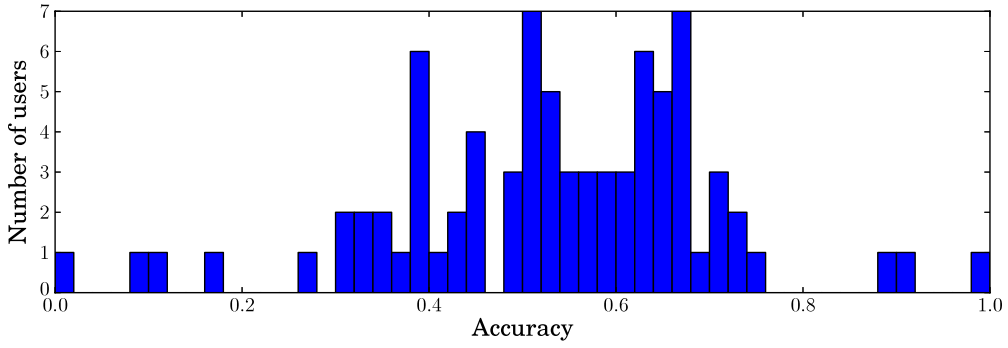


Fig. 8. Histogram of the accuracy on set *C*, for all users. For each user, we chose the best predictor on set *B*. This figure shows that there is a high variance in the predictability of users.

Despite the fundamental differences between the three families of predictors, they exhibit very close average prediction accuracies and outperform significantly the baseline predictors. However, when looking at users individually, the performance of each family varies greatly, as illustrated in Fig. 7 for three selected users. We explain these variations by the fact that users have different types of behavior that are captured, with different levels of faithfulness, by each family of predictors.

Moreover, as shown in Fig. 8, we observe a high variance of the predictability of users: we reach a prediction accuracy of 100% for the most predictable user, whereas we predict correctly 0% of the time for the least predictable one. Besides the intrinsic unpredictability of people, the major factor causing such a poor prediction performance is the lack of data: the data available about the user, for which we make no correct predictions, span over a period of only 12 days.

We obtained these results by using only basic features of the visits. In an attempt to improve the accuracy of our predictors, we include additional contextual information, such as distance between places, GSM cell towers, WLANs or accelerometer data, but we observed no improvement. We also implemented various preprocessing techniques, such as clustering and feature embedding, with no improvement either.

5. Blending

As expected, the accuracies of the predictors presented in Section 4 are not equal, but more importantly, each predictor makes different errors: samples for which a predictor fails might be those on which another excels, as illustrated below in Section 5.2. This idea is the foundation of blending, in which we combine several predictors, in order to take advantage of their diversity.

5.1. Notation

Before describing each blending strategy, we first define Φ, the set of all predictors trained on the data. This set can be split into three subsets of predictors $\Phi = \Phi_{\text{DBN}} \cup \Phi_{\text{ANN}} \cup \Phi_{\text{GBDT}}$, where each subset corresponds to all predictors of one same family. For instance, Φ_{GBDT} is the set of all predictors using the GBDT method.

A predictor ϕ is defined by its family, some internal parameters, and the data it was trained on. Thus, we can refer to the predictor ϕ in general, or to a specific predictor $\phi(u)$, that was trained using the data of a user u. We do not mention the dependence on u when it is obvious from the context.

Please cite this article in press as: V. Etter, et al., Where to go from here? Mobility prediction from instantaneous information, Pervasive and Mobile Computing (2013), http://dx.doi.org/10.1016/j.pmcj.2013.07.006

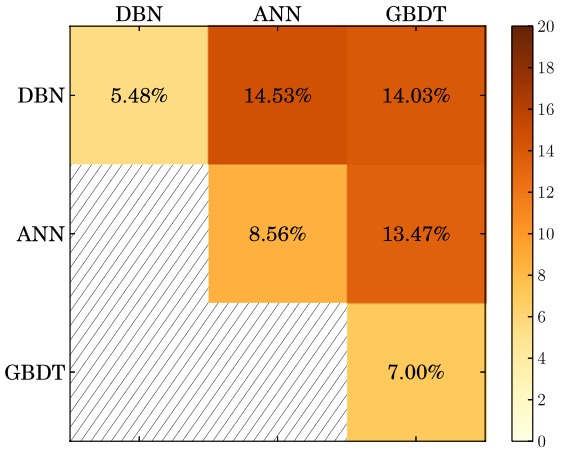


Fig. 9. Diversity between the best 10 predictors of each family, measured as the average proportion of samples for which, given a random pair of predictors, only one of them predicts the correct next place. The diagonal shows the diversity between predictors of the same family, as defined in (6), whereas the other values show the diversity between two families of predictors, as defined in (7). Predictors of different families clearly show a higher diversity than predictors of the same family, which suggests that blending all three families would result in a better accuracy than individual families.

5.2. Diversity of predictors

We evaluate the diversity between two predictors by the proportion, averaged over all users, of samples for which *only* one of them predicts the correct next place. Therefore, the diversity between two predictors ϕ_1 and ϕ_2 is defined as

$$\text{diversity}(\phi_1, \phi_2) = \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} \frac{1}{N_u} \sum_{n=1}^{N_u} \left| \mathbb{I}_{\{\hat{X}_n^{\phi_1}(u) = X_{n+1}(u)\}} - \mathbb{I}_{\{\hat{X}_n^{\phi_2}(u) = X_{n+1}(u)\}} \right|,$$

where $\mathcal{U}$ is the set of all users, N_u is the number of samples for user u, $\hat{X}_n^{\phi}(u)$ is the prediction made by the predictor ϕ for the nth sample of user u, as defined in (1), and $X_{n+1}(u)$ is the true next place.

To measure the diversity of a set of predictors Φ, we compute the average diversity between all possible pairs:

$$\text{diversity}(\Phi) = \frac{1}{|\Phi|(|\Phi| - 1)} \sum_{\phi_1 \in \Phi} \sum_{\phi_2 \in \Phi \setminus \phi_1} \text{diversity}(\phi_1, \phi_2). \tag{6}$$

Similarly, we measure the diversity between two sets of predictors Φ_1 and Φ_2 by computing the average diversity between all possible pairs:

$$\text{diversity}(\Phi_1, \Phi_2) = \frac{1}{|\Phi_1|\,|\Phi_2|} \sum_{\phi_1 \in \Phi_1} \sum_{\phi_2 \in \Phi_2} \text{diversity}(\phi_1, \phi_2). \tag{7}$$

Fig. 9 shows the diversity of our predictors, both intra- and inter-family. To avoid taking into account predictors that have a poor prediction performance, which would bias the diversity measure, we only keep the 10 best predictors of each family, according to their performance on set B, and compare their predictions over set C.

With fewer than 6% of samples for which only one predictor out of a random pair of is correct, the DBN family is the most consistent of the three families. However, we see that DBN and ANN have more than 14% of such samples, even though they have similar performances overall. This suggests that blending these families together will result in increased performance, which is confirmed in the results presented in Section 5.4.

5.3. Blending strategies

Below, we briefly explain the five strategies we submitted to the NMDC. The accuracy of each predictor on a given set is computed using (2).

Strategy 1. For each user, we choose the predictor that has the best accuracy on set B:

$$\phi_1 = \arg\max_{\phi \in \Phi} \{A_B(\phi)\}.$$

Strategy 2. For each user, the predictor is a weighted mixture of all predictors, where the weight of each predictor is proportional to its average performance on set B:

$$\phi_2 = \frac{1}{\sum_{\phi \in \Phi} A_B(\phi)} \sum_{\phi \in \Phi} A_B(\phi) \cdot \phi.$$

Please cite this article in press as: V. Etter, et al., Where to go from here? Mobility prediction from instantaneous information, Pervasive and Mobile Computing (2013), http://dx.doi.org/10.1016/j.pmcj.2013.07.006

Strategy 3. For each user, we first select the best predictor of each family:

$$\phi_{\text{DBN}} = \arg\max_{\phi \in \Phi_{\text{DBN}}} \{A_B(\phi)\},$$
$$\phi_{\text{ANN}} = \arg\max_{\phi \in \Phi_{\text{ANN}}} \{A_B(\phi)\},$$
$$\phi_{\text{GBDT}} = \arg\max_{\phi \in \Phi_{\text{GBDT}}} \{A_B(\phi)\}.$$

Then, we simply combine these three predictors uniformly:

$$\phi_3 = \frac{1}{3}\phi_{\text{DBN}} + \frac{1}{3}\phi_{\text{ANN}} + \frac{1}{3}\phi_{\text{GBDT}}.$$

Strategy 4. As we have many predictors (3 families with a large space of parameters), a majority of them have an average performance. Thus, when we blend them together, the few good predictions made by the best models are averaged out by all the other predictions. This is particularly true for Strategy 2. To prevent this and still guarantee some diversity, we first select the top 10 predictors of each family:

$$\Phi_{\text{top}} = \{\phi : \phi \text{ is one of the best 10 predictors of its family}\}.$$

The final predictor is a mixture of this subset of predictors, weighted by their performance on set B:

$$\phi_4 = \frac{1}{\sum_{\phi \in \Phi_{\text{top}}} A_B(\phi)} \sum_{\phi \in \Phi_{\text{top}}} A_B(\phi) \cdot \phi.$$

Strategy 5. We choose the predictor that has the best average accuracy over all users:

$$\phi_5 = \arg\max_{\phi \in \Phi} \left\{ \sum_{u \in \mathcal{U}} A_B(\phi(u)) \right\},$$

where $\mathcal{U}$ is the set of all users, and $\phi(u)$ corresponds to the predictor ϕ trained using the data of user u. Contrarily to the others, this strategy chooses the same predictor for all users.

We could also use non-linear blenders like neural networks, where we learn the optimal combination of the individual predictors. We could even go further and use sample-based blending techniques, where we adapt the combination of the blenders to the features of each sample. However, due to the limited amount of data available in the NMDC, we limit ourselves to linear blenders. Indeed, more sophisticated blending techniques would require larger validation sets.

5.4. Results

Fig. 10 shows the resulting accuracies on set C of the blending strategies described above. We have empirical evidence that making use of the diversity of the predictors, while taking into account their individual performances, increases significantly the prediction accuracy. For example, the most successful blending strategy (Strategy 4) is a mixture of the 10 best predictors of each family where the contribution of each predictor is proportional to its accuracy. This strategy outperforms Strategy 1 where we take simply the most accurate predictor for each user (55.55% vs. 53.17%, *i.e.*, a relative improvement of 4%).

These observations are corroborated by the prediction accuracies on the undisclosed set D (also shown in Fig. 10), which were revealed by the organizers at the end of the challenge. They confirm the accuracies measured on set C: the ranking of the strategies relative to their accuracy is respected, and Strategy 4 is still the best with a prediction accuracy of 56.22%.

6. Related work

With the increasing availability of human mobility datasets comes a growing scientific interest in studying human mobility and in understanding the mechanisms that govern it. The literature is composed of both descriptive and predictive approaches: the descriptive approach [3,11–14] is based on modeling both individual and group mobility. The main goal of the descriptive approach is to capture the statistical properties of human mobility and to ensure, for the sake of realism, that the trajectories generated by mobility simulators exhibit these same properties. The predictive approach [15–17], however, focuses on the implementation of methods that predict accurately the locations users will visit in the near future. Naturally, the approach we take is predictive, as the main goal of our present work is to predict as accurately as possible the next place a user will visit. In this section, we present a selection of articles that we believe are representative of the rich literature about human mobility prediction.

Song et al. [3] study the predictability of human mobility using a dataset of 45,000 mobile phone users. They try to answer the fundamental question, "To which extent is human mobility predictable?" They represent the mobility of each user as

Please cite this article in press as: V. Etter, et al., Where to go from here? Mobility prediction from instantaneous information, Pervasive and Mobile Computing (2013), http://dx.doi.org/10.1016/j.pmcj.2013.07.006

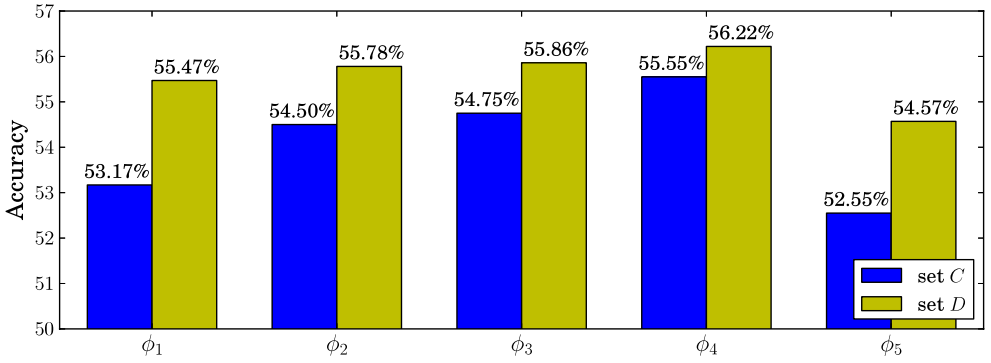


Fig. 10. Accuracy on sets C and D of the five different blending strategies, described in Section 5.3.

the sequence of detected cell towers. They quantify the users' mobility predictability by approximating the entropy rate of their mobility process (process generating a sequence of cell tower ids). They find out that, on average, 1 bit of information is needed to describe the next cell tower a user will visit. They claim that human are 93% predictable, and that the predictability varies slightly across the whole population, which suggests that we are all equal in predictability. However, the authors do not implement a mobility predictor to verify empirically the claims presented.

Song et al. [15] evaluate the performance of several location predictors using a two-year trace of the mobility of over 6000 users. They represent the mobility of a user as the sequence of Wi-Fi access points detected. The authors claim that the major challenges faced when it comes to mobility prediction are the unseen contexts and the sudden change of users' habits. To overcome these drawbacks, the authors enhance their predictors with fallback and aging mechanisms, resulting in an enhanced prediction accuracy. Their best predictor is a second-order Markov chain with fallback mechanism and has a median prediction accuracy of about 72%.

Similarly, Scellato et al. [18] use a non-linear method to predict the time and duration of a user's next visit to one of his significant places. Their method identifies patterns in a user's mobility history that are similar to his recent movements in order to predict his behavior. Note that, as stated in Section 2, we cannot use high-order methods for the Next-Place Prediction challenge because we have access to information about the present visit only.

In order to enhance the classic approach to mobility prediction, Cho et al. [16] study the influence of the social dimension on mobility: they claim that human mobility is a combination of periodic movements and seemingly random jumps that are correlated with the social network of the user. They develop a mobility model based on these observations and evaluate its performance on a mobility dataset composed of GPS points and cell tower ids. The results do not show a systematic improvement of prediction accuracy when the social dimension is taken into account. However, the authors expect that, with denser datasets, the social dimension will bring significant improvement to mobility prediction.

The related work we have introduced and, more generally, the studies on human mobility, rely heavily on empirical evidences: the authors analyze a mobility dataset in order to find interesting patterns, capture statistical properties or test the methods they implemented. The temptation is to draw from this analysis a conclusion about human mobility and its fundamental properties (distribution of distance between consecutive locations visited, predictability, etc.) without taking into consideration the characteristics of the dataset studied and their impact on the results found. For example, quantifying the predictability of human mobility depends strongly on the resolution of location information available: predicting the next cell tower a user will visit could be straightforward [3] but finding his exact location within this cell is much more challenging.

7. Conclusion

In this paper, we present various mobility predictors for the Nokia Mobility Data Challenge. We use a wide range of techniques, including Probabilistic Graphical Models and Artificial Neural Networks. Moreover, we adapt these techniques to the characteristics of the data, by implementing various mechanisms that ensure the adaptability of the predictors to the sudden changes in users' behavior and the sparsity of the data. In particular, the aging techniques allow us to improve significantly the prediction accuracy for the users with an important change of habits (home change, start of a new semester for students, relationship breakup, etc.).

In order to benefit from the diversity of these predictors, we also introduce several blending strategies that combine them into a global and more accurate predictor. Despite the simplicity of these techniques, the predictors – obtained after blending – are able to bring a relative improvement of up to 4% over individual predictors. Each of the five blending strategies we submitted to the NMDC outperformed all the other participants' submissions, allowing our team to be the winner of the Next-Place Prediction challenge.

Our predictors reach an average prediction accuracy of more than 56%, yet we observe a high variance between users. Is this unpredictability mainly rooted in the users' personality, or is it a consequence of the data characteristics?

Please cite this article in press as: V. Etter, et al., Where to go from here? Mobility prediction from instantaneous information, Pervasive and Mobile Computing (2013), http://dx.doi.org/10.1016/j.pmcj.2013.07.006

References

[1] J.K. Laurila, D. Gatica-Perez, I. Aad, J. Blom, O. Bornet, T.-M.-T. Do, O. Dousse, J. Eberle, M. Miettinen, The mobile data challenge: big data for mobile computing research, in: Mobile Data Challenge by Nokia Workshop, Springer, Newcastle, UK, 2012.
[2] N. Kiukkonen, J. Blom, O. Dousse, D. Gatica-Perez, J. Laurila, Towards rich mobile phone datasets: lausanne data collection campaign, in: Proceedings of the ACM International Conference on Pervasive Services (ICPS), Berlin, ACM, 2010.
[3] C. Song, Z. Qu, N. Blumm, A.-L. Barabási, Limits of predictability in human mobility, Science 327 (2010) 1018–1021.
[4] C.M. Bishop, Pattern Recognition and Machine Learning, first ed., in: Information Science and Statistics, Springer, 2007, 2006. Corr. 2nd printing edition.
[5] L. Song, D. Kotz, R. Jain, X. He, Evaluating next-cell predictors with extensive Wi-Fi mobility data, Technical Report, Dartmouth College, 2006.
[6] R. Collobert, Torch, NIPS Workshop on Machine Learning Open Source Software, 2008.
[7] Y. Le Cun, L. Bottou, G.B. Orr, K.-R. Müller, Efficient backprop, in: Neural Networks, Tricks of the Trade, in: Lecture Notes in Computer Science LNCS, vol. 1524, Springer Verlag, 1998.
[8] N. Morgan, H. Bourlard, Generalization and parameter estimation in feedforward netws: some experiments, in: Advances in Neural Information Processing Systems 2, Morgan Kaufmann, 1989, pp. 630–637.
[9] J.H. Friedman, Stochastic gradient boosting, Computational Statistics & Data Analysis 38 (2002) 367–378.
[10] J.H. Friedman, Greedy function approximation: a gradient boosting machine, The Annals of Statistics 29 (2001) 1189–1232.
[11] M.C. Gonzalez, C.A. Hidalgo, A.-L. Barabási, Understanding individual human mobility patterns, Nature 453 (2008) 779–782.
[12] C. Song, T. Koren, P. Wang, A.-L. Barabási, Modelling the scaling properties of human mobility, Nature Physics 6 (2010) 818–823.
[13] I. Rhee, M. Shin, S. Hong, K. Lee, S.J. Kim, S. Chong, On the levy-walk nature of human mobility, IEEE/ACM Transactions on Networking 19 (2011) 630–643.
[14] M. Kim, D. Kotz, S. Kim, Extracting a mobility model from real user traces, in: Proceedings of INFOCOM 2006. 25th IEEE International Conference on Computer Communications, IEEE Computer Society Press, 2006, pp. 1–13.
[15] L. Song, D. Kotz, R. Jain, X. He, Evaluating next-cell predictors with extensive Wi-Fi mobility data, IEEE Transactions on Mobile Computing 5 (2006) 1633–1649.
[16] E. Cho, S.A. Myers, J. Leskovec, Friendship and mobility: user movement in location-based social networks, in: Proceedings of the 17th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, ACM, 2011, pp. 1082–1090.
[17] D. Ashbrook, T. Starner, Using GPS to learn significant locations and predict movement across multiple users, Personal and Ubiquitous Computing 7 (2003) 275–286.
[18] S. Scellato, M. Musolesi, C. Mascolo, V. Latora, A.T. Campbell, Nextplace: a spatio-temporal prediction framework for pervasive systems, in: K. Lyons, J. Hightower, E.M. Huang (Eds.), Pervasive Computing, in: Lecture Notes in Computer Science, vol. 6696, Springer, Berlin, Heidelberg, 2011, pp. 152–169.

Please cite this article in press as: V. Etter, et al., Where to go from here? Mobility prediction from instantaneous information, Pervasive and Mobile Computing (2013), http://dx.doi.org/10.1016/j.pmcj.2013.07.006