Computer Communications 000 (2016) 1–15

Contents lists available at ScienceDirect

Computer Communications

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

MobHet: Predicting human mobility using heterogeneous data sources

Lucas M. Silveira[a,*], Jussara M. de Almeida[a], Humberto T. Marques-Neto[b], Carlos Sarraute[c], Artur Ziviani[d]

[a] Universidade Federal de Minas Gerais – UFMG, Brazil
[b] Pontifical Catholic University of Minas Gerais – PUC Minas, Brazil
[c] Grandata Labs, Argentina
[d] National Laboratory for Scientific Computing – LNCC, Brazil

ARTICLE INFO

Article history:
Available online xxx

Keywords:
Human mobility prediction
Georeferenced data
Mobile phone data

ABSTRACT

The literature is rich in mobility models that aim at predicting human mobility. Yet, these models typically consider only a single kind of data source, such as data from mobile calls or location data obtained from GPS and web applications. Thus, the robustness and effectiveness of such data-driven models from the literature remain unknown when using heterogeneous types of data. In contrast, this paper proposes a novel family of data-driven models, called MobHet, to predict human mobility using heterogeneous data sources. Our proposal is designed to use a combination of features capturing the popularity of a region, the frequency of transitions between regions, and the contacts of a user, which can be extracted from data obtained from various sources, both separately and conjointly. We evaluate the MobHet models, comparing them among themselves and with two single-source data-driven models, namely SMOOTH and Leap Graph, while considering different scenarios with single as well as multiple data sources. Our experimental results show that our best MobHet model produces results that are better than or at least comparable to the best baseline in all considered scenarios, unlike the previous models whose performance is very dependent on the particular type of data used. Our results thus attest the robustness of our proposed solution to the use of heterogeneous data sources in predicting human mobility.

1. Introduction

The analysis of human mobility patterns can assist in the design of various mechanisms to help improving the quality of services that support urban life [1]. For example, the knowledge of typical patterns of human mobility within a target geographic area, such as a large urban center, can drive the management and planning of transportation routes to minimize the chance of congestion during heavy traffic periods and the planning of urban occupation, as well as support decisions towards a faster contention of the spread of a disease [1,2]. Moreover, such knowledge may also support a more cost-effective planning of the infrastructure of mobile phone networks [3], thus improving the quality of service offered by carriers. Understanding human mobility within a particular region is also a key step towards building human mobility prediction models, which can support the design of various services, such as location-based recommendation systems [4–6].

Focusing specifically on human mobility prediction, recent studies have shown that human mobility in urban areas can be fairly predictable considering daily routines [5,7]. Yet, most mobility prediction models available in the literature [8–11] have been proposed or at least evaluated considering only a single type of data, such as data associated with mobile phone calls or georeferenced data collected from GPS and web applications. The use of different types of data from heterogeneous sources, however, can lead to more accurate mobility predictions or at least to a larger coverage of the population [12]. Although some previous prediction models, proposed specifically in the context of location recommendation, do exploit data from different sources, such sources are often different Web applications (e.g., Twitter and Foursquare) [13], which share common aspects (e.g., mobility inferred from user check-ins). To the best of our knowledge, no previous study investigated the prediction of human mobility by combining data from as heterogeneous sources as mobile phone calls and online social networks (e.g., Twitter).

In this context, this article presents a novel family of data-driven models, called MobHet, to predict human mobility using heterogeneous data sources. The design of the MobHet models combines principles and assumptions explored by two existing

* Corresponding author.
E-mail address: lucasmaiasilveira@gmail.com (L.M. Silveira).

http://dx.doi.org/10.1016/j.comcom.2016.04.013
0140-3664/

mobility prediction models that we use as baselines, namely Leap Graph [8] and SMOOTH [9]. Notably, MobHet exploits the popularity of different regions within the target geographic area, a feature also used by SMOOTH, as well as the frequency of transitions across different regions, the main feature used by Leap Graph. Moreover, previous studies have observed that the contacts among people may influence their movement patterns [11,14,15]. That is, a person may be influenced by whom he/she knows is going to a certain place. In this case, human mobility prediction models that consider relationships which demonstrate prior knowledge between users (e.g., friendship) may produce better results than a model that disregard such information [16]. Motivated by this observation, some MobHet models may also exploit the (locations of) contacts of a person to predict his/her location, an aspect that was neglected by the aforementioned baseline methods. By exploring different combinations of these three features—popularity of a region, frequency of transitions between regions, and contacts among people—we define the proposed MobHet models that aim at predicting where a person will be located in a given future time.

A key aspect that distinguishes the MobHet models from prior work is the type of data source used to support the human mobility predictions. Unlike the baseline models, MobHet was designed to use data from heterogeneous sources, both separately and conjointly. We here show the use of the MobHet models with mobile phone data and data collected from a online social network. However, in principle, any type of data from which the considered features can be extracted (or inferred) could be used.

We evaluate the prediction accuracy of MobHet models, comparing them with themselves and with both SMOOTH and Leap Graph, the two single-source data-driven models from which MobHet was inspired, in various scenarios consisting of: (i) homogeneous data from a single source, for different types of data, notably mobile phone call logs or GPS location data collected from Twitter; and (ii) the combination of data from both sources. The evaluation is performed using a large set of different real-world datasets collected in different locations and time periods, involving hundreds of thousands to millions of users depending on the dataset, with a number of events (mobile phone calls or posted tweets) in the same order of magnitude.

Our experimental results indicate that our best MobHet model, which exploits all three aforementioned features conjointly, produces the most accurate predictions, being at least comparable to or superior than the best baseline, in all evaluated scenarios. In contrast, the performance of both SMOOTH and Leap Graph is very dependent on the particular type of data used. SMOOTH achieves results comparable to our best model using Twitter location data, but presents much lower performance in the other scenarios, whereas Leap Graph obtains results similar to our best solution for mobile phone data (for which it was designed) and also when the data from the two sources are combined. Yet, the perfomance of Leap Grap falls behind both SMOOTH and MobHet when only Twitter location data is used.

In short, the contributions of this paper are: (i) the proposal of a family of mobility prediction models that exploit combinations of features as well as different heterogeneous sources of data; and (ii) a thorough evaluation of our models, as well as of two baseline models—SMOOTH and Leap Graph—in different scenarios with homogeneous and heterogeneous data sources while considering different time windows for mobility inference. Overall, our evaluation attests the robustness of the proposed MobHet solution to the use of heterogeneous data in predicting human mobility.

The remainder of this article is organized as follows. Section 2 formally states the problem of human mobility prediction we tackle, discusses related work on human mobility models, and explains the operation of the two models adopted as baselines. Section 3 introduces our new MobHet models. The data sets used in the experiments and the adopted evaluation methodology are presented in Section 4. Section 5 discusses our main experimental results, while conclusions and possible directions for future work are presented in Section 6.

Table 1
Adopted notation to model the human mobility prediction.

Variable	Definition
U	Set of users
R	Set of regions
T	Set of time windows
$\mathcal{D}^{training}$	Training set composed of tuples $< u_i, r_j, t_k >$, where $u_i \in U$, $r_j \in R$, and $t_k \in T$, used to learn the human mobility patterns and to derive the prediction models
$\mathcal{D}^{test}$	Test set composed of tuples $< u_i, r_j, t_k >$, where $u_i \in U$, $r_j \in R$, and $t_k \in T$, used to evaluate the prediction models
p_j	Popularity of the region $r_j \in R$, learned from $\mathcal{D}^{training}$
$h_{j1,\,j2}$	Frequency of transitions between r_{j1} and r_{j2}, learned from $\mathcal{D}^{training}$
$w_{j1,\,j2}$	Weight of edge (r_{j1}, r_{j2}), learned from $\mathcal{D}^{training}$
d	Parameter used to define the size of a region: it may be the size of the side, in the case of square regions, or the radius in the case of circular regions
f_{dist}	Distribution of distances traveled by a user $u_i \in U$ during a movement
f_{pause}	Distribution of time intervals between successive movements of a user $u_i \in U$ (i.e., pause times)
t_{max}	Max number of time windows in $\mathcal{D}^{training}$
C_i	Set of contacts of user $u_i \in U$, learned from $\mathcal{D}^{training}$
$L_{i,\,j,\,k}$	Event user u_i is located in region r_j during time window t_k
m	The minimum number of contacts of all users
$C^{\geq m}_{i,j,k}$	Event at least m contacts of user u_i are located in region r_j during time window t_k
$ninteractions_{i1,\,i2}$	Number of times user u_{i1} interacted with u_{i2}
$ContactStrength_{i1,\,i2}$	Strength of the contact between u_{i1} and u_{i2}, from the perspective of u_{i1}
θ	Threshold used to determine the list of contacts of a user, according to the *ContactStrength* approach

2. Background

We start this section by first introducing the human mobility prediction task we aim at addressing in this paper (Section 2.1). Next, we briefly discuss previous related studies (Section 2.2), delving further into the presentation of two reference mobility prediction models, namely SMOOTH and Leap Graph, which are used as baselines in our experimental evaluation (Section 2.3). Table 1 presents the main notation used in this section as well as throughout the paper.

2.1. Problem statement

The prediction task we tackle in this article can be defined as follows. *Given a target geographic area, defined of a set R of non-overlapping regions $r_j \in R$, a set of users $u_i \in U$ and a set of time windows $t_k \in T$, we want to build a model to predict where (in which region r_j) a given user u_i will be at a future time window t_k.*

To tackle this prediction problem we assume that a training set $\mathcal{D}^{training}$ as well as a test set $\mathcal{D}^{test}$ are given: each such set consists of tuples $< u_i, r_j, t_k >$, indicating that u_i was in region r_j during time window t_k. Moreover, both sets *may* contain data from which we can infer that a relationship demonstrating prior knowledge (e.g., friendship) between users u_i and u_j ($u_i \neq u_j$) exists. We refer to such relationship as a *contact* between u_i and u_j, and we define the set of contacts of user u_i as C_i. We want to learn the prediction model, possibly inferring the set of contacts of each user

Please cite this article as: L.M. Silveira et al., MobHet: Predicting human mobility using heterogeneous data sources, Computer Communications (2016), http://dx.doi.org/10.1016/j.comcom.2016.04.013

u_i, using only data in the training set $\mathcal{D}^{training}$. We want to evaluate the learned model using the test data, that is, we want to use the model to predict the location (i.e., region) r_j for each tuple $< u_i$, ?, $t_k >$ in the test set $\mathcal{D}^{test}$.

Note that the definition of regions r_j can be done in several ways. For example, each region may be defined by a center point x_j, y_j (e.g., the coordinates of an antenna in case of mobile phone data) and a radius d. Alternatively, each region may be defined by a squared area with center x_j, y_j and side d (total area d^2). We here adopt the latter definition by dividing the total target area into non-overlapping squared regions r_j, as a grid.

Also note that, in this article, we discretize the total time interval T in consecutive fixed time windows t_k for the purpose of determining the movements of each user. In other words, the location of a user is predicted considering the granularity of each window t_k.

The split of the data into training and test sets can also be performed in different ways, depending on the data available. Regardless, the split should respect time constraints, i.e., training data must precede the test data chronologically (i.e., $\forall < *, *, t_{k1} > \in \mathcal{D}^{training}$ and $\forall < *, *, t_{k2} > \in \mathcal{D}^{test}$, $t_{k1} < t_{k2}$)[1]. Moreover, considering that human mobility patterns vary through the day or even on different days (e.g., weekdays and weekends) [3], the training data should reflect time periods comparable to those covered in the test set. For example, if we wish to predict the location of users between 8AM and 9AM, the training data used to build such model should have been collected during that same period in previous days or in a period short before the target one on the same day (so that the assumption of similar mobility patterns still hold). The definition of the training and test sets in our experiments will be discussed in Section 4.1.

2.2. Related work

The literature is rich in solutions that aim at supporting human mobility prediction. The existing methods exploit different types of data as well as various strategies to try to predict, as accurately, as possible, the future location of people (i.e., users) [17]. Some of these models aim at predicting the *trajectory* of users using GPS data [18] or mobile phone data [8]. Others make the predictions using statistical distributions that capture specific patterns extracted from the data, such as the distribution of the distances traveled by a user during a movement [9,10,16,19].

Other studies, such as [20] and [21], explore various patterns observed in the data, such as the places visited by people, the frequency of the visits and the regularity of a user mobility patterns. In [22], the authors explore the places visited as well as the distances covered by a user within the same region to predict the user location at a future time. In [14], the authors address the prediction mobility considering a combination of short-ranged paths and long distance travels. They find that while the former is periodic, both spatially and temporally, the latter is more influenced by the user's social network ties.

Similarly, using GPS data collected from georeferenced social networking applications, both Davis et al. [23] and Nguyen et al. [24] propose mobility prediction models that exploit not only the distances traveled and the locations visited by users, but also the friendship relations between users (extracted directly from the data). In other words, the models assume that a person will have more chance to go to a place where his/her friends often visit. Some other prior efforts infer such friendship relations from the data and use them to study the relationship between friendship and mobility. Alharbi et al. [25] created a model to predict users' social ties from the locations visited by them. The assumption explored by the authors is that two people who visit the same location at the same time interval can be considered friends. Scellato et al. [13] take a step further and assume that a social bond exists between two users if they are located in *nearby* locations.

In common, all those previous models of human mobility have been proposed or evaluated considering a single type of data, either data associated with mobile phone calls [2,8,26] or GPS data [9,14,19,24,27–29]. Although some prior efforts explored data from different sources, they often combine similar data (e.g., check-ins) from different Web applications (e.g., Twitter and Foursquare) [4,13]. The robustness of these models when configured using different type of data, either separately or jointly, is still unknown. Moreover, the recent study by Hess et al. [12] show how the use of heterogeneous sources may lead to more accurate mobility predictions or at least to a larger coverage of the population.

This observation motivated us to design the MobHet models, introduced in Section 3. MobHet is actually a family of models that can use multiple data sources to learn mobility patterns and predict the future location of users. MobHet inherits some of its features from two reference models, namely SMOOTH [9] and Leap Graph [8]. Nevertheless, MobHet goes beyond both reference models, by also exploiting the relationships (or contacts) among people to predict mobility, as done by other prior efforts [23,24]. Moreover, unlike SMOOTH, Leap Graph and the other models that exploit user contacts, MobHet can be configured with different types of data as input.

This work greatly extends a preliminary effort of ours to develop human mobility prediction models [30]. In our prior work, we proposed a single model that exploits only two sets of features, namely region popularity and the frequency of transitions between different regions. In this article, we go much further by proposing a family of models, which differ in terms of the features they exploit: in addition to the two aforementioned features, our models also make use of the relationships (contacts) among the users. By comparing all proposed models, we are able to assess the extent to which each such feature is important for mobility prediction, which has not been analyzed before. Moreover, we evaluate different strategies to infer user contacts from the data, as will be further discussed in Section 3.2. Finally, we here evaluate our models much more thoroughly than in [30], covering more datasets and scenarios.

Before introducing our new MobHet models, we first describe how our two baseline models, SMOOTH and Leap Graph, work next.

2.3. Baseline models

This section describes the main components of the two reference models adopted in this work, SMOOTH [9] and Leap Graph [8]. In our experiments, we used as reference the implementations of both models made available by the authors[2]. In order to to enable the comparison of the models, we did few code changes, e.g. in Leap Graph. These changes will be explained in Section 2.3.2.

2.3.1. SMOOTH

The SMOOTH model [9] was originally designed to reproduce patterns of human mobility and not to predict the future location of a user. However, after analyzing the model source code made available by its authors, we noticed pieces of code (and associated

[1] The chronological split is motivated by the goal of evaluating the prediction model in a realistic scenario where only historical data, collected at previous time periods, is available to build the prediction model.

[2] SMOOTH: http://toilers.mines.edu/Public/Code/smooth.html ; Leap Graph: https://www.cs.utexas.edu/~wdong86/. We had accessed both source codes on October 28th, 2015.

comments) that enable SMOOTH to predict user locations. Therefore, for comparison purposes, we considered in this paper this particular implementation of SMOOTH as available in the original code. Moreover, the original model captures the movements of a group of users in a simulated two-dimensional area consisting of a set of circular regions. Yet, for the sake of a fair comparison, we used the same region definition based on a grid for all considered models. That is, we assumed a simulated area which is divided into a set of non-overlapping squared regions. Each region r_j is defined by the coordinates x_j, y_j of its center and a side d.

The main assumption of SMOOTH is that people tend to move towards popular regions. Thus, each region r_j has an associated probability p_j that a user will move towards it. The probability p_j captures the popularity of the region r_j, that is, the expected number of people visiting r_j, and this probability is extracted from the training set.

The basic idea of SMOOTH is to simulate the movement of the users in U in a sequence of steps. Each step corresponds to a time window t_k, using two distributions extracted from the training set: the distribution of distances traveled by a user during a movement f_{dist} and the distribution of pause times f_{pause} between successive movements.

The simulation works as follows. At each step, for each user who is not paused (explained below), we first compute the direction of the movement in relation to the user's current location and the probabilities associated with each region $r_j \in R$, and then randomly select a distance using f_{dist}. We then simulate the movement of the user using both the direction and distance selected to determine where (i.e., in which region) the user will be in the next time window. Then, we randomly pick a pause time using f_{pause}, and simulate the user staying in the same location during the selected time. After each step, the probabilities associated with each region are recomputed.

Thus, the training of the model is done in two stages. The first consists of extracting distributions f_{dist} and f_{pause} as well as the initial set of regions R and their probabilities p_i from the training set $\mathcal{D}^{training}$. All regions that have users in $\mathcal{D}^{training}$ are initially introduced in R with the corresponding probabilities. In the second stage of the training, the initial positions of the users are determined from the probabilities associated with the regions, and the movements of each user are simulated using distributions f_{dist} and f_{pause} for a number of time windows equal to t_{max} (i.e., the number of time windows in $\mathcal{D}^{training}$). Note that new regions may be discovered during this phase (i.e., regions in the grid with no user in the training set). These new regions are inserted into set R, with their associated probabilities.

During the test phase, the movements of the users in set $\mathcal{D}^{test}$ are simulated using the model learned during training, keeping the set R fixed and considering the starting location of each user given by the user's first appearance in $\mathcal{D}^{test}$. The regions visited by users during the simulation are compared with the data available in $\mathcal{D}^{test}$ to assess the accuracy of the predictions.

2.3.2. Leap Graph

In [8], the authors investigated how to use mobile phone data to predict the mobility of users and proposed a new mobility prediction model named Leap Graph. They assumed as input a log of phone calls, each one characterized by the following information: unique user identifier, the time instants of the beginning and end of the call, the coordinates (latitude and longitude) of the antennas where the call started and ended, and the identifiers of the sectors used in those antennas[3].

In its original implementation, Leap Graph assumes that each region $r_j \in R$ is associated with an antenna, being defined by the antenna's coordinates and a radius d corresponding to the antenna's coverage area. However, for the sake of a fair comparison, we here assume the same region definition for all models, based on a grid, as defined in the previous section[4]. Thus, calls are associated with the regions where the used antennas are located. That is, a call from user u_i associated with an antenna located within region r_j at time t is interpreted as an evidence of the presence of u_i in that region during the time window t_k that includes t. The model attempts to infer the movement of each user from a graph that captures the trajectories of users across the regions of R.

The training phase consists primarily in creating a graph of trajectories for each user using the data in $\mathcal{D}^{training}$. In graph g_i created for user u_i, each vertex corresponds to a region. An edge between r_{j1} and r_{j2} is added when: (i) u_i made a call that started in r_{j1} and terminated in r_{j2}; or when (ii) u_i made two consecutive calls, the first one in r_{j1} and the second in r_{j2}.

As originally proposed, Leap Graph aims to predict the next region where a user will be located given its current location. Therefore, it does not consider the time dimension and only explores the transitions between regions made by each user. To make it comparable to SMOOTH and to our proposed MobHet models (which consider user movements over time) and to apply it to our target prediction task, we added self-loops to each vertex to capture the staying of a user in the same region in successive time windows. We also assigned weights $w_{j1,\,j2}$ to each edge (r_{j1}, r_{j2}), including self-loops, to capture the probability of a user moving towards the destination region r_{j2} (or staying in the same region, in case of a self-loop) in the next time window.

To compute such weights we make two key assumptions: (1) the time interval between two successive calls from the same user is equally divided between the two regions where the calls took place[5], and (2) a transition between two regions occur in a single time window. Specifically, suppose that user u_i made two consecutive calls, one during time window t_{k1} while located in region r_{j1}, and the next one during time window t_{k2} while in region r_{j2}. From this data, we are not able to infer how long u_i remained in r_{j1} (which is important to compute the weight of the self-loop associated with r_{j1}) before moving to r_{j2} nor how long such movement took. Thus, we assume that: (1) the user remained half of the time interval between t_{k1} and t_{k2} in r_{j1} and the other half in r_{j2}, and (2) the transition between the two regions happened during one time window. This is obviously an approximation, but it was a design choice based on the available data.

Given such assumptions, we first compute the weight of each edge (including self-loops), $w_{j1,\,j2}$, as the total number of times the user performed the corresponding transition. In the case of self-loops, the weight corresponds to the total number of time windows during which the user remained in the corresponding region. Putting in other words, the weight of a self-loop is equal to the number of times a transition to the same region was performed, assuming transitions occur during one window. Let us consider the same example of a user who made two calls, one during t_{k1} while in r_{j1} and the next one in t_{k2} while in r_{j2}. In this case, we would assign the weights of the two self-loops as $w_{j1,\,j1} = w_{j2,\,j2} = \frac{t_{k2} - t_{k1}}{2}$, and the weight of the edge between r_{j1} and r_{j2} as $wj1,\,j2 = 1$. This computation is performed for each edge of each graph g_i. Note that, given such design choice, we are not able to capture trips

[3] Each antenna is divided into 3 sectors of 120^o, each one responsible for about one third of the antenna coverage area.

[4] We note however that preliminary experiments with the same region definition as in the original papers of both SMOOTH and Leap Graph (i.e., circular regions) led to similar conclusions in terms of the relative performance of all models in all considered scenarios as those reported in this article.

[5] For calls that started and ended in different regions, we assume that the duration of the call is equally divided between the two regions.

Please cite this article as: L.M. Silveira et al., MobHet: Predicting human mobility using heterogeneous data sources, Computer Communications (2016), http://dx.doi.org/10.1016/j.comcom.2016.04.013

starting and ending within the same time window. That is, we capture time differences at the granularity of time windows.

As proposed in [8], the individual graphs created are then combined into a single weighted graph G which captures the movements of the population of users in $\mathcal{D}^{training}$. To that end, the user graphs g_i are sorted in an order given by the time of the first call of each user in $\mathcal{D}^{training}$ and then processed as follows. The edges of all graphs are combined into G. Each edge weight is recomputed to represent the average over all individual graphs. Moreover, at the end of the training phase, the weights of all edges leaving the same vertex (including self-loops) are normalized to add up to 1, so as to represent probabilities of a user performing the transition (or remaining in the same region, in case of self loops) in one time window.

One particular issue regarding the combination of individual graphs in G deserves further explanation. Dong et al. [8] consider that users who visit the same sequence of regions tend to continue following the same route. That is, if two graphs capturing the trajectories of two users have one or more paths covering the same edges, with at least n (an input parameter) edges in common, the two trajectories are considered "similar", and the edges of the second graph following the common path are disregarded. As an example, suppose that graph g_1 contains the trajectory $\{r_1, r_2, r_3, r_4\}$ and graph g_2 contains the trajectory $\{r_1, r_2, r_3, r_5\}$, and suppose parameter n is set to 2. Given that trajectory $\{r_1, r_2, r_3\}$, with $n = 2$ edges, appears in both graphs, the edges present in g_2 following the common path (i.e, (r_3, r_5)) would be disregarded. Graph G would contain the edges that build trajectory $\{r_1, r_2, r_3, r_4\}$. The weights of edges (r_1, r_2) and (r_2, r_3) would be equal to 2 (prior to normalization), and the weight of (r_3, r_4) would be set to 1. Thus, parameter n specifies a minimum number of edges in common that two graphs should have to be considered "similar" trajectories and to cause edges of the second graph to be disregarded. According to the authors, this measure is taken to avoid double counting the same trajectory, since they only consider the path and not the volume of users who traverse between regions.

During the test phase, Leap Graph simulates the movement of users using graph G. The initial location of each user is defined from the user's first call in $\mathcal{D}^{test}$. This corresponds to a vertex in graph G. We use the edge weights as probabilities to simulate the movements of each user and infer his/her location in successive time windows. In our experiments, we set the number n of edges equal to 2, as this produced the best results in [8].

3. The new MobHet models

In this section, we introduce a new family of data-driven human mobility prediction models, named MobHet. Unlike prior models, MobHet exploits a combination of the following features: popularity of regions, the frequency of transitions among regions, and the contacts of each user. In the following, we first explain how MobHet works (Section 3.1), and then we discuss different strategies applied to define the contacts of a user (Section 3.2).

3.1. MobHet operation

The MobHet model[6] exploits heterogeneous data sources, either jointly or individually, to capture the movement of users among regions r_j in R. Once again, we assume the total area simulated is divided into squared regions r_j (with center x_j, y_j and a side d) as a grid.

MobHet inherits some aspects of the two baseline models. As done in SMOOTH, we associate a popularity measure with each region r_j. Similarly to Leap Graph (and unlike SMOOTH), MobHet simulates the movement of users between regions with a region transition graph G, with self-loops representing the permanency of a user in the same region in successive time windows. However, unlike Leap Graph, the creation of graph G does not start with the graphs of individual trajectories for each user, but rather considers all regions visited by all users jointly. The weight of an edge (including self-loop) is computed by exploiting a combination of the three features, region popularity, frequency of transition between regions, and user contacts, as discussed below.

In MobHet, the popularity of a region r_j, p_j, is given by the number of users who visited r_j in the training set $\mathcal{D}^{training}$ (may be 0), while the frequency of transition between regions r_{j1} and r_{j2}, denoted by $h_{j1,\,j2}$, is given by the *average* number of times the transition was performed in consecutive time windows by the same user (also in $\mathcal{D}^{training}$). Note that, like in Leap Graph, the weight of a self-loop captures the average number of time windows a user remains in the same region, as this is interpreted as successive transitions from/to the same region. Moreover, just like in Leap Graph, we assume that the time between successive pieces of evidence of the user location (e.g., successive calls, successive tweets) is equally divided between the two associated regions, and that the movement between the two regions occurs in a single time window.

Motivated by previous observations that a user's social links may influence her/his movements [13,16], we also exploit the *contacts* among users in the design of MobHet. A contact between two users can be defined as any interaction held between them that indicates some prior relationship. This is a very broad notion, which can be instantiated in different ways, depending on the data available from which such contacts will be extracted. We discuss the adopted strategies to define the contacts of a user in Section 3.2.

Given the lists of contacts of each user, we define the probability $P(L_{i,j,k}|C^{\geq m}_{i,j,k})$ of finding user u_i in region r_j in time window t_k (event $L_{i,\,j,\,k}$), given that at least m contacts of user u_i are located in the same region in the same time window (event $C^{\geq m}_{i,j,k}$). That is

$$P(L_{i,j,k}|C^{\geq m}_{i,j,k}) = \frac{P(L_{i,j,k} \ \wedge \ C^{\geq m}_{i,j,k})}{P(C^{\geq m}_{i,j,k})},$$

As the other two features, namely region popularity (p_j) and frequency of transition between regions ($h_{j1,\,j2}$), the probabilities $P(L_{i,j,k}|C^{\geq m}_{i,j,k})$ are computed using the training set $\mathcal{D}^{training}$.

In its present form, MobHet builds a single graph that captures an aggregated behavior of the whole user population[7]. For example, when exploiting user contacts to build such graph (as discussed below), we take the mean probability $P(L_{i,j,k}|C^{\geq m}_{i,j,k})$ across all users. Similarly, region popularity and frequency of transitions capture the interests for particular regions and movements across regions of the whole user population. We chose to do so for the sake of simplicity and to have a first-cut estimate of the benefits of capturing (a subset of) the three aforementioned features for mobility prediction, even in an aggregated way. However, we note that MobHet could be extended to employ other strategies. For instance, we could compute the considered features across clusters of users with similar mobility patterns (learned from historical data), thus building different graphs, one of each cluster. This approaches would be more costly, but could potentially lead to better results. We leave those extensions and a thorough evaluation of their benefits for future work.

[6] Details about the implementation of MobHet, including source code, as well as our implementation of the (modified) Leap Graph, can be found in http://homepages.dcc.ufmg.br/~lucasmsil/Mobhet.html.

[7] Note that both SMOOTH and Leap Graph also capture aggregated behavior of all users.

Given the three sets of features, we define a family of predictive models, called MobHet. Specifically, we define the following variations of MobHet:

1. *Transition and popularity (MobHet-TP)*: exploits only the frequency of transitions between regions and the popularity of each region. In this case, the weight of edge (r_{j1}, r_{j2}) is defined $w_{j1,j2} = h_{j1,j2} \times p_{j2}$ for any edge (i.e., $j1 \neq j2$ or $j1 = j2$). That is, in this case, the probability $P(L_{i,j2,k}|C^{\geq m}_{i,j2,k})$ is disregarded;
2. *Transition and contacts (MobHet-TC)*: exploits only the frequency of transitions between regions and the contacts of each user. In this case, the weight of edge (r_{j1}, r_{j2}) is $w_{j1,j2} = h_{j1,j2} \times P(L_{i,j2,k}|C^{\geq m}_{i,j2,k})$ for any edge (i.e., $j1 \neq j2$ or $j1 = j2$);
3. *Popularity and contacts (MobHet-PC)*: the weight of edge (r_{j1}, r_{j2}) is $w_{j1,j2} = p_{j2} \times P(L_{i,j2,k}|C^{\geq m}_{i,j2,k})$ for any edge (i.e., $j1 \neq j2$ or $j1 = j2$);
4. *Only contacts (MobHet-C)*: the weight of edge (r_{j1}, r_{j2}) is $w_{j1,j2} = P(L_{i,j2,k}|C^{\geq m}_{i,j2,k})$ for any edge (i.e., , $j1 \neq j2$ or $j1 = j2$);
5. *Transition, popularity, and contacts (MobHet-TPC)*: exploits all three features jointly. In this case, the weight of edge (r_{j1}, r_{j2}) is given by $w_{j1,j2} = h_{j1,j2} \times p_{j2} \times P(L_{i,j2,k}|C^{\geq m}_{i,j2,k})$ for any edge (i.e., $j1 \neq j2$ or $j1 = j2$).

In all cases, the weights are normalized to represent transition probabilities (i.e., the sum of the weights of all edges leaving a vertex must add up to 1).

MobHet thus differs from both Leap Graph, which considers only the transitions between regions, and from SMOOTH, which exploits only the popularity of each region. Our family of mobility models considers that both aspects as well as the contacts of a user may influence the trajectory of the user: (i) on one hand, users tend to visit specific locations depending on his/her current location (as shown in [8]); (ii) on the other hand, the popularity of a region [9,29], as well as the location of the contacts of the user [24] may also influence where the user goes next. By comparing the aforementioned five variations of MobHet against themselves as well as with both SMOOTH and Leap Graph, we are able to assess the extent to which each of the three features is important to improve prediction accuracy, either in isolation (e.g., MobHet-C, SMOOTH) or combined (e.g., MobHet-TPC, MobHet-TP). We discuss the results of this evaluation in Section 5.

The training phase of each MobHet model consists in learning from the training set the values of the three used features and build the transition graph G. We illustrate this process for the MobHet-TP model in Fig. 1. In Fig. 1(a), the vertices are labeled with the region popularity values (e.g., $p_2 = 5$ for r_2) and the edges are labeled with the frequency of transitions (e.g., $h_{1,2} = 4$ for edge (r_1, r_2)). The edge weights ($w_{j1,j2}$) are first computed as the product of both measures (Fig. 1(b)) and then normalized (Fig. 1(c)). Note that, like Leap Graph, we are not able to capture trips starting and ending within the same time window.

The test phase consists of simulating graph G, as done for Leap Graph. One final issue refers to the use of the MobHet models that exploit user contacts (i.e., MobHet-C, MobHet-PC, MobHet-TC, and MobHet-TPC) in cases where we cannot infer the location of enough contacts of the user. In that case, we revert to the corresponding MobHet variation without contacts. For example, suppose we are using MobHet-TPC to make predictions. If a prediction is to be made for a user u_i who does not have at least m contacts in D^{test}, we then choose to use MobHet-TP instead. Similarly, we use MobHet-P (MobHet-T) for users with not enough contacts, instead of MobHet-PC (MobHet-TC). Therefore, during test phase, we keep always a pair of transition graphs, switching between them depending on whether the target user has enough contacts or not. If MobHet-C is the model in use, only predictions for users with at least m contacts in D^{test} can be made.

3.2. User contacts

Having described the general operation of MobHet, we now examine how to define a contact of a user. As mentioned above, a contact between two users can be defined as any interaction between them that can serve as evidence that such users know each other (virtually or in the real world). For example, a phone call involving two users can be used as evidence that these users know each other, thus implying that they are contacts of each other. Similarly, a social link between two users in a Web application, such as a follower-followee connection on Twitter, can also be used to define the contacts between users.

We here evaluate two strategies to define the contacts of a user. The first one, which has already been addressed in the literature [13,21,22,28], uses the friendship ties present in online social networks, being suitable to be used when data from such applications is available. Specifically, we here use the follower-followee links on Twitter. That is, if user u_1 *follows and is followed by* user u_2, they are considered contacts of each other. We refer to this strategy as *Follower-Followed*.

The second strategy, referred to as *ContactStrength*, explores the frequency of interactions between two users, and can be applied to data collected from both social networks (Twitter, in the present case) and mobile phone calls. Specifically, we consider as an interaction either a phone call or a retweet, and define the *strength* of the contact between a pair of users u_1 and u_2 as follows. Let $ninteractions_{i1,\,i2}$ be the number of times user u_{i1} interacted with u_{i2}, either by posting a *retweet* or by making/receiving a call from/to u_{i2}. Note that $ninteractions_{i1,\,i2}$ captures the total number of interactions between the two users, regardless of who initiated the call or posted the retweet. The strength of the contact between u_{i1} and u_{i2} *from the perspective of* u_{i1} is defined as the fraction of all interactions of u_{i1} which happened with u_{i2}. That is

$$ContactStrength_{i1,i2} = \frac{ninteractions_{i1,i2}}{\sum_{i3=1,i3\neq i1}^{|U|} ninteractions_{i1,i3}},$$

We consider that u_{i2} is a contact of u_{i1} if $ContactStrength_{i1,\,i2}$ is equal to or greater than a given threshold θ. In Section 5, we present an evaluation of the impact of the choice of θ for prediction accuracy.

Note that, unlike in the *Follower-Followed* strategy, the contact links in the *ContactStrength* approach are *not* necessarily bidirectional. That is, u_{i2} may be considered a contact of u_{i1} even if u_{i1} is *not* a contact of u_{i2}.

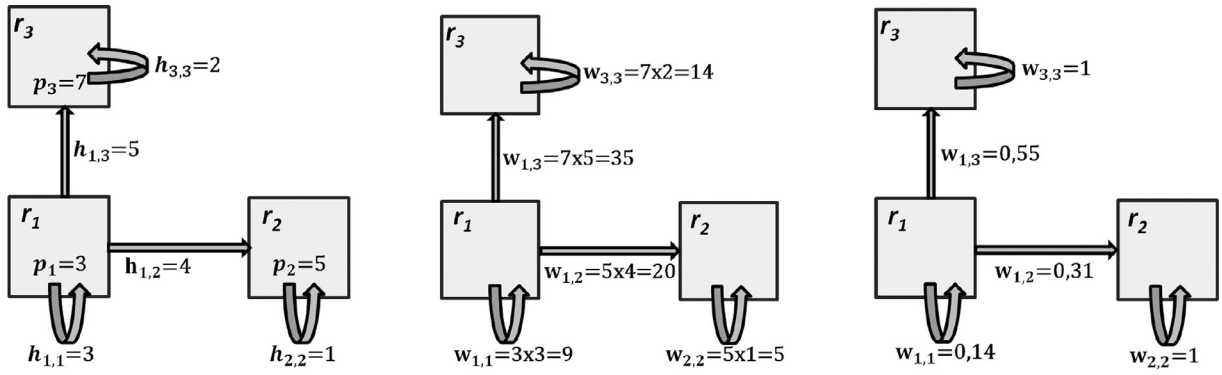
4. Experimental setup

In this section, we present the datasets (Section 4.1) as well as the methodology (Section 4.2) used in our experimental evaluation of the human mobility prediction models.

4.1. Datasets

In our evaluation, we use three different datasets obtained from multiple sources. The first dataset consists of data related to mobile phone calls collected in different major cities in Brazil during different periods of time. This dataset, referred to as MobilePhone-BR, is further described in Section 4.1.1. The second dataset, referred to as Twitter-BR and detailed in Section 4.1.2, consists of georeferenced tweets collected at the same locations and during the same periods as the MobilePhone-BR dataset. Finally, our third dataset consists of mobile phone calls collected in Mexico during a one-month period. This dataset, described in Section 4.1.3, is referred to as MobilePhone-MX. At the end of this section, we

Please cite this article as: L.M. Silveira et al., MobHet: Predicting human mobility using heterogeneous data sources, Computer Communications (2016), http://dx.doi.org/10.1016/j.comcom.2016.04.013



(a) Popularity of regions and frequency of transitions

(b) Computing the edge weights

(c) Normalization of edge weights

Fig. 1. Determining transition probabilities for MobHet-TP: an illustrative example.

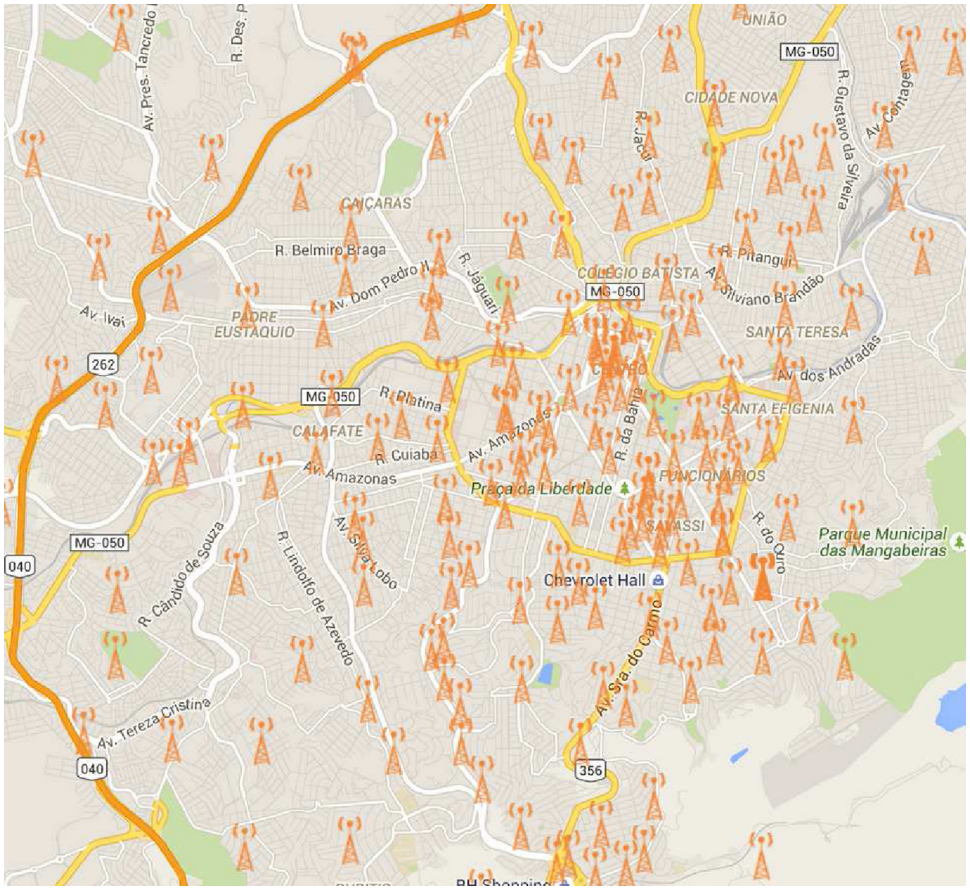


Fig. 2. Locations of the Antennas in Belo Horizonte city (MobilePhone-BR dataset).

discuss some filtering applied to the data and provide an overview of all datasets (Section 4.1.4).

4.1.1. MobilePhone-BR dataset

Our first dataset, provided by a large Brazilian mobile phone operator, is composed of information about mobile phone calls made during pre-specified time periods in five major cities in Brazil, namely Belo Horizonte (BH), Fortaleza, Recife, Rio de Janeiro (RJ), and São Paulo (SP).

The data contains the following information for each call:

- *Call id*: unique identifier of the call;
- *User id*: unique identifier of the user who made the call (anonymized);
- *Start time*: start time of the call;
- *End time*: end time of the call;
- *Initial antenna*: geographic coordinates (*i.e.*, latitude and longitude) of the antenna where the call was initiated;

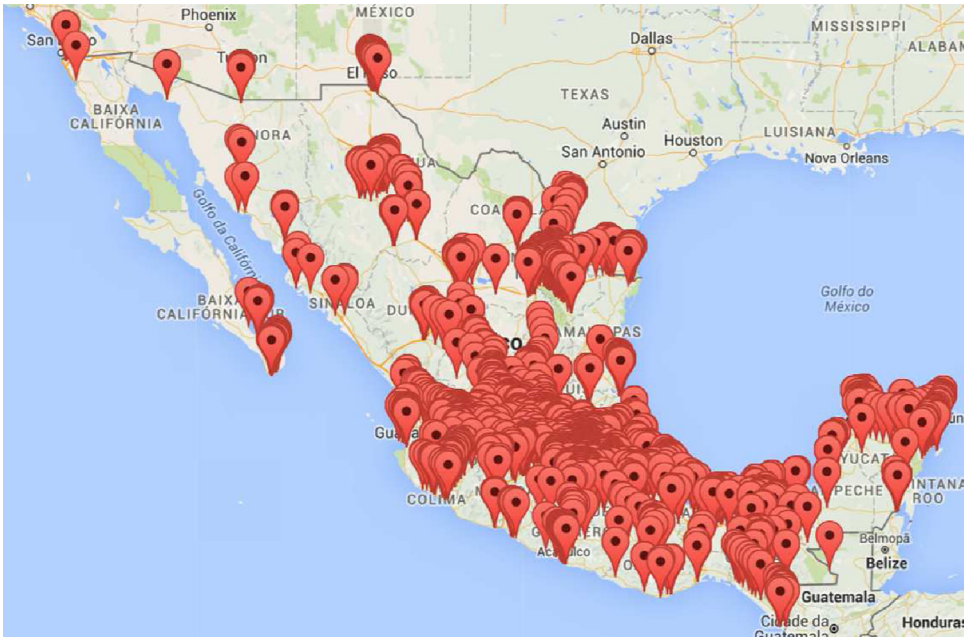


Fig. 3. Locations of the antennas in Mexico (MobilePhone-MX dataset).

- *End antenna*: geographic coordinates (*i.e.,* latitude and longitude) of the antenna where the call was finished.

For illustration purposes, Fig. 2 shows the locations of the antennas in the city of Belo Horizonte[8]. An overview of the amount of data available for each city is given in Section 4.1.4.

4.1.2. Twitter-BR dataset

The Twitter data consists of georeferenced tweets, *i.e.,* tweets with geographic coordinates, collected using the Twitter *Stream API*. This API allows real-time gathering of tweets with *location* filtering, thus enabling the restriction of the collection area to a particular region. The collection of the Twitter dataset was planned following the same locations and time periods on which the MobilePhone-BR was gathered. Such locations and periods were agreed upon with the mobile phone operator beforehand, prior to the effective data gathering. Each registered tweet contains the following information:

- *Tweet id*: unique identifier of the tweet;
- *User id*: unique identifier of the user who posted the tweet (anonymized);
- *Latitude*: latitudinal geographic coordinate from where the user posted the tweet;
- *Longitude*: longitudinal geographic coordinate from where the user posted the tweet;
- *Time*: timestamp of when the user posted the tweet;
- *Retweets*: list of users (identified by their user ids) who posted retweets of this tweet.

Besides that, for each user u_i, we also collected

- *Followers*: list of users (identified by their user ids) that follow u_i;
- *Followed*: list of users (identified by their user ids) followed by u_i.

Note that both the MobilePhone-BR and the Twitter-BR datasets have been collected *independently*. Therefore, it is not possible to identify the same user on these different data sources. In this way, we consider both sets of users disjoint.

4.1.3. MobilePhone-MX dataset

In addition to the MobilePhone-BR and Twitter-BR datasets collected in Brazil (see Section 4.1.1 and 4.1.2, respectively), we also evaluated our prediction models using a mobile phone dataset provided by Grandata[9]. This dataset consists of information about phone calls made in 22,304 antennas spread over Mexico (Fig. 3) during a one-month period (March 1st to 31st, 2014). The data is anonymized and contains the following information for each call:

- *Call id*: unique identifier of the call;
- *User id*: unique identifier of the user who made the call (anonymized);
- *Destination id*: unique identifier of the user who received the call (anonymized);
- *Start time*: start time of the call;
- *End time*: end time of the call;
- *Initial antenna*: geographic coordinates (*i.e.,* latitude and longitude) of the antenna where the call was initiated;
- *End antenna*: geographic coordinates (*i.e.,* latitude and longitude) of the antenna where the call was finished.

The single difference between the structure of the MobilePhone-BR (Section 4.1.1) and the MobilePhone-MX datasets is that in the latter the anonymized user id to whom the call was destined is also known.

4.1.4. Filtering and resulting dataset sizes

We applied a filtering to all three datasets to remove users who made only one call or posted only one tweet, since we could not infer any mobility of such users from a single event (tweet or call). Table 2 summarizes the resulting datasets collected in Brazil (MobilePhone-BR and Twitter-BR) after the application of this filtering. Each row shows the location (city) and the collection

[8] The map was taken from the site Telebrasil: http://www.telebrasil.org.br/panorama-do-setor/mapa-de-erbs-antenas.

[9] http://www.grandata.com.

Table 2
Individual collections that compose MobilePhone-BR and Twitter-BR datasets.

Brazilian city	Date	Time	Calls		*Tweets*	
	MM/DD/YYYY	interval	#Calls	#Users	#Tweets	#Users
Belo Horizonte (BH)	10/21/2011	13 h–21 h	31.705	12.237	32.334	11.231
Belo Horizonte (BH)	12/31/2011	20 h–04 h	201.212	100.021	210.001	105.000
Belo Horizonte (BH)	01/03/2012	20 h–04 h	12.145	5.246	40.234	17.342
Belo Horizonte (BH)	02/03/2013	13 h–20 h	69.227	30.033	30.765	10.338
Belo Horizonte (BH)	03/10/2013	13 h–20 h	15.794	7.585	27.340	12.845
Belo Horizonte (BH)	03/02/2013	12 h–19 h	15.630	9.354	14.332	4.870
Belo Horizonte (BH)	06/22/2013	13 h–21 h	4.050	1.998	30.103	12.540
Belo Horizonte (BH)	06/26/2013	13 h–21 h	6.264	2.987	29.934	11.532
Belo Horizonte (BH)	09/11/2013	17 h–23 h	14.023	4.532	15.635	5.103
Fortaleza	06/29/2014	14 h–21 h	7.185	2.372	13.453	4.236
Recife	12/31/2011	20 h–04 h	21.123	10.000	45.321	20.192
Recife	01/03/2012	20 h–04 h	8.769	4.390	7.839	2.987
Recife	06/29/2014	14 h–21 h	13.335	4.923	13.577	3.981
Rio de Janeiro (RJ)	08/28/2011	14 h–20 h	67.627	28.027	38.091	13.227
Rio de Janeiro (RJ)	10/30/2011	14 h–20 h	58.610	25.593	37.931	12.498
Rio de Janeiro (RJ)	12/04/2011	14 h–20 h	77.869	30.597	39.239	12.945
Rio de Janeiro (RJ)	12/11/2011	14 h–20 h	56.159	23.563	40.123	13.002
Rio de Janeiro (RJ)	12/31/2011	20 h–04 h	36.354	13.918	21.021	3.211
Rio de Janeiro (RJ)	01/03/2012	20 h–04 h	20.231	9.134	45.322	19.443
Rio de Janeiro (RJ)	03/29/2012	18 h–22 h	31.166	12.305	45.030	15.302
Rio de Janeiro (RJ)	07/08/2012	14 h–20 h	7.579	3.384	30.213	13.490
Rio de Janeiro (RJ)	11/27/2013	18 h–00 h	17.009	6.192	32.940	13.834
Rio de Janeiro (RJ)	06/29/2014	14 h–21 h	5.120	1.132	14.033	3.643
Rio de Janeiro (RJ)	07/13/2014	14 h–21 h	5.340	1.038	15.860	4.572
São Paulo (SP)	02/04/2012	15 h–22 h	3.370	1.159	25.370	11.930
São Paulo (SP)	11/25/2012	12 h–18 h	22.752	11.235	28.042	13.220
São Paulo (SP)	03/24/2013	13 h–20 h	44.499	20.787	50.323	20.334
Total			874.147	383.742	974.406	392.848

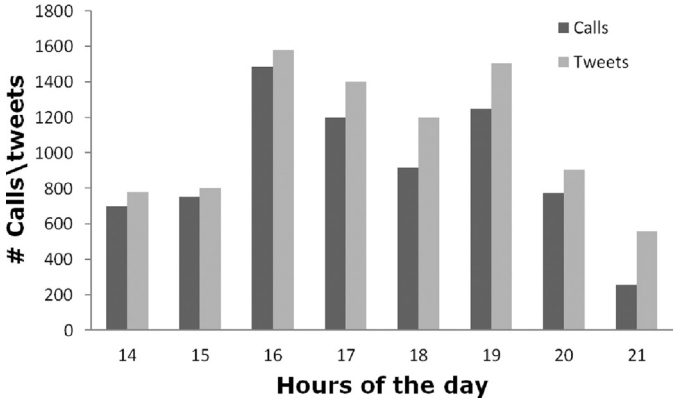


Fig. 4. Amount of calls/tweets per hour–Rio de Janeiro 06/29/14.

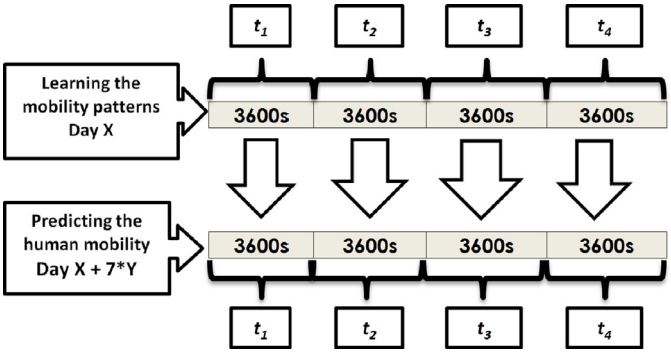


Fig. 5. Separation of data into training and test sets (different days).

period, the number of calls and tweets as well as the respective numbers of users in each dataset. In total, the datasets collected in Brazil cover five major cities, 27 days, 874,147 calls made by 383,742 users, and 974,406 tweets posted by 392,848 users. The filtered dataset from Mexico (MobilePhone-MX), in turn, consists of a total of 9882,477 calls made by 3541,580 users throughout the month of March, 2014.

4.2. Evaluation methodology

Before discussing our evaluation methodology, we argue that, as one might expect, the volume of data (i.e., number of calls or tweets) varies greatly over the day. This is illustrated in Fig. 4 for one particular city (Rio de Janeiro) and period (June 29th, 2014) in the MobilePhone-BR dataset. Thus, we decided to develop a predictive model for every hour in order to more accurately capture the underlying mobility patterns in different periods during the day. To that end, we first divided each dataset into one-hour intervals.

Next, we divided each dataset into training ($\mathcal{D}^{training}$) and test ($\mathcal{D}^{test}$) sets. As argued in Section 2.1, such division should be made provided that the following assumption holds: the training set, where the prediction model is learned, covers time periods during which the mobility patterns are similar to those present in the test set, where the model is applied. The first approach is to use data from multiple days (same day of the week but different weeks) covering the same period of time. In that case, we could use the data in one day for training, and the data in the same weekday, one week later, for testing the model as illustrated in Fig. 5. For this strategy we use the following datasets (MM/DD/YYYY is the date format):

- Belo Horizonte 12/31/2011 and Belo Horizonte 01/03/2012;
- Belo Horizonte 02/03/2013 and Belo Horizonte 03/10/2013;
- Recife 12/31/2011 and Recife 01/03/2012;
- Rio de Janeiro 08/28/2011 and Rio de Janeiro 10/30/2011;
- Rio de Janeiro 12/04/2011 and Rio de Janeiro 12/11/2011;
- Rio de Janeiro 12/31/2011 and Rio de Janeiro 01/03/2012;
- Rio de Janeiro 06/29/2014 and Rio de Janeiro 07/13/2014;

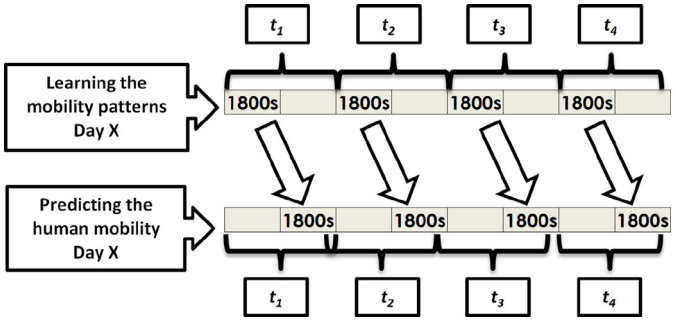


Fig. 6. Separation of data into training and test sets (same day).

- Mexico 16 to 22 of March of 2014 and Mexico 23 to 29 of March of 2014.

However, some collections in our Brazilian dataset do not cover multiple days. Thus, we opted for a different strategy applying it to all datasets. For each day, we further divided each one-hour interval into two 30-minute periods: the former was used for training, and the latter for testing, as illustrated in Fig. 6.

An important aspect of the evaluation is the definition of the time window and time period. The time window captures the mobility of users during the time period used in the datasets to train and test the models. The time period, in contrast, is used to break the timeline into intervals during which (we believe) human mobility will be roughly stable. Since the time period of the collections are 30 min (for training and test at the same day) and 60 min (for training and test in different days), we decide to use time windows of 5 and 10 min. Thus, we observe at least three steps for periods of 30 min and six steps for the period of 60 min. Further, we use a *null* model in which user u_i located in a region r_j at time t_k remains in the same region at time t_{k+1}.

For each of the datasets collected in Brazil, we evaluated each mobility prediction model in scenarios that are based on only mobile phone calls, only tweets, and on both mobile phone calls and tweets simultaneously. The purpose of the latter scenario is to evaluate the performance of the models when configured to use heterogeneous data sources together. The best approach to combine data from heterogeneous sources is not obvious because different models may have different relative performances depending on the input data. Thus, we consider and evaluate two strategies to combine mobile phone calls and tweets:

- Association of tweets to mobile phone calls: each tweet is associated with the nearest cell antenna to the tweet geolocation;
- Association of mobile phone calls to tweets: each antenna in the mobile phone call dataset is considered a point (as is each tweet) in the simulation region.

Note that in both cases, we are not able to infer user contacts from the combined data as this information is not available on the mobile phone calls. Thus, when using heterogeneous data, we only evaluate the MobHet variation that does not exploit contacts (MobHet-TP).

As explained in Section 2.1, 2.3 and 3.1, we used the same definition for regions for the sake of comparability. Additionally, we defined the simulated area of the Brazilian datasets as the geographic region of the city where each dataset was collected. For the Mexican dataset, we used the geographic area of the country as input to simulation. In all cases, we consider the distance d defining each region as equivalent to 500 m, which is the typical coverage radius of an antenna. This value was chosen so that the results of the various scenarios are comparable.

Finally, we evaluated the *prediction accuracy* for all models, that is defined as the fraction of tuples $< u_i, r_j, t_k >$ in the test set $\mathcal{D}^{test}$ for which the prediction was correct. The only reason for having the training data to precede the test data chronologically is that we want to simulate a realistic scenario where, at prediction time, we only have historical data collected over previous periods of time.

5. Experimental results

In this section, we turn to evaluate the new MobHet models, comparing them with themselves and with the two baselines, SMOOTH and Leap Graph. We start by comparing all MobHet variations, introduced in Section 3.1, with each other (Section 5.1). Next, we analyze different scenarios for our model (Section 5.2). Finally, we compare our best MobHet models with the two baselines (Section 5.3).

5.1. Accuracy of the MobHet models

The five variations of MobHet model differ in terms of the three basic features—region popularity, frequency of transitions between regions, and user contacts—used to compute the weights of the edges of the region transition graph G. In this section, we focus on comparing these five variations, namely, MobHet-C, MobHet-TP, MobHet-TC, MobHet-PC, and MobHet-TPC, verifying their respective prediction accuracy. Here, we evaluate the models using only the datasets Twitter-BR and MobilePhone-MX because as aforementioned we cannot infer user contacts from the MobilePhone-BR dataset. We use this latter dataset for comparing our best model with the baselines.

For the Twitter-BR dataset, we apply two strategies to identify user contacts, namely *Follower-Followed* and *ContactStrength* (described in Section 3.2), while for the MobilePhone-MX dataset we only use the *ContactStrength* strategy. For both datasets, we evaluate all models that use contacts the following values of θ (the *ContactStrength* threshold): 10%, 15%, 20%, 25%, 50%, and 75%. For these models, we initially fix the value of m, the minimum number of contacts considered, at 1, deferring the evaluation of the impact of this parameter to the end of this section.

Table 3 shows the average prediction accuracy (along with 95% confidence intervals) of each of the five proposed MobHet models and various contact definition strategies[10]. The table shows results for one single city and time period of the Twitter-BR dataset (Rio de Janeiro, 06/29/2014 with time window of 5 min and training and test at the same day). We omitted the results for other periods/cities because they are very similar. Best results (and statistical ties) for each dataset and model are shown in bold, whereas the overall best result for each dataset is marked with a "*".

Overall, the best results are produced by MobHet-TPC. This shows the importance of considering all three features jointly to predict human mobility. The worst approach is the one that uses only contacts: MobHet-C produces results that are as much as 48% (Twitter-BR) and 24% (MobilePhone-MX) worse than MobHet-TPC, besides not being applicable to users with no contacts in the dataset. Next, using both contacts and frequency of transitions between regions (MobHet-TC) or region popularity (MobHet-PC) greatly improves over using only the former. Yet, both approaches are still worse than MobHet-TPC. Finally, it is interesting to note that using both region popularity and frequency of transitions (MobHet-TP) improves over using only contacts (up to 59% and 32% for Twitter-BR and MobilePhone-MX datasets, respectively). Yet, this approach is still much worse than MobHet-TC, MobHet-PC and MobHet-TPC, which indicates the importance of taking the user contacts into account when predicting mobility, consistently

[10] Each model was replicated 50 times for all experiments present in the Section 5.

Table 3
Average precision (along with 95% confidence intervals) of all MobHet models ($m = 1$). Best results (and statistical ties) for each method are shown in bold. Overall best results are marked with "*".

Twitter-BR Dataset (Rio de Janeiro, 06/29/2014)					
Contact definition		*MobHet-C*	*MobHet-TC*	*MobHet-PC*	*MobHet-TPC*	*MobHet-TP*
Follower-Followed		0.342 ± 0.0194	0.505 ± 0.0135	0.513 ± 0.0176	0.559 ± 0.0185	
Contact Strength	$\theta = 10\%$	0.365 ± 0.0175	0.540 ± 0.0142	0.567 ± 0.0182	0.625 ± 0.0195	
	$\theta = 15\%$	0.373 ± 0.0170	0.560 ± 0.0147	0.594 ± 0.0180	0.640 ± 0.0190	
	$\theta = 20\%$	0.383 ± 0.0195	0.595 ± 0.0151	0.632 ± 0.0174	0.654 ± 0.0184	
	$\theta = 25\%$	**0.409 ± 0.0149**	**0.620 ± 0.0169**	**0.653 ± 0.0177**	**0.678 ± 0.0193***	
	$\theta = 50\%$	**0.420 ± 0.0159**	**0.630 ± 0.0173**	**0.653 ± 0.0162**	**0.684 ± 0.0197***	
	$\theta = 75\%$	0.394 ± 0.0184	0.612 ± 0.0163	0.621 ± 0.0149	0.631 ± 0.0192	
None						**0.545 ± 0.0211**
MobilePhone-MX Dataset						
Contact definition		*MobHet-C*	*MobHet-TC*	*MobHet-PC*	*MobHet-TPC*	*MobHet-TP*
Contact Strength	$\theta = 10\%$	0.493 ± 0.0232	0.593 ± 0.0191	0.613 ± 0.0193	0.630 ± 0.0201	
	$\theta = 15\%$	0.509 ± 0.0222	0.602 ± 0.0195	0.626 ± 0.0182	0.649 ± 0.0213	
	$\theta = 20\%$	0.524 ± 0.0214	0.623 ± 0.0180	0.642 ± 0.0190	0.668 ± 0.0210	
	$\theta = 25\%$	**0.544 ± 0.0230**	**0.641 ± 0.0198**	**0.663 ± 0.0198**	**0.692 ± 0.0172***	
	$\theta = 50\%$	**0.532 ± 0.0218**	**0.638 ± 0.0208**	**0.652 ± 0.0203**	**0.683 ± 0.0168***	
	$\theta = 75\%$	0.468 ± 0.0212	0.575 ± 0.0212	0.583 ± 0.0215	0.618 ± 0.0178	
None						**0.619 ± 0.0153**

Table 4
Number of users in $D^{training}$ as we vary parameter θ of *ContactStrength* approach.

ContactStrength	DataSet	
	Twitter-BR (Rio de Janeiro on 06/29/2014)	MobilePhone-MX
$\theta = 10\%$	692	1332,480
$\theta = 15\%$	621	1265,381
$\theta = 20\%$	553	1203,072
$\theta = 25\%$	400	1032,182
$\theta = 50\%$	392	1005,387
$\theta = 75\%$	78	232,321

with previous studies [13,16]. As an example, the improvements of MobHet-TPC over MobHet-TP vary from 3% to 26% on the Twitter-BR dataset, and from 2% to 12% on the MobilePhone-MX dataset, depending on the specific strategy employed to infer user contacts.

Regarding these strategies, we note that exploiting the strength of the contacts is much better (up to 27%) than using the follower and followee links on Twitter. This is not surprising, as the latter captures a weaker relationship between the two users, and thus, reflects potentially less influence of one user on the other. Interestingly, we find that consistently for all methods and both datasets, the best results obtained with the *ContactStrength* approach are produced for θ equal to either 25% or 50%. Smaller values of θ lead to less conservative contact strategies, less accurate contact inferences, and, ultimately, less accurate predictions. As the value of θ increases, the inferences become more reliable and prediction accuracy improves. Yet, very large values of θ (e.g., $\theta = 75\%$) impose serious constraints on the size of the training set, as the number of users who have enough contacts in $D^{training}$ is reduced.

The reduction on the amount of data from which the model is learned impacts its ability to generalize, ultimately hurting its accuracy on the test set. As shown in Table 4, the number of unique users in the training set decreases as θ increases on both datasets. However, the reduction is quite sharp when θ goes from 50% to 75%, which may cause the drop in performance of the MobHet models.

The results discussed above were obtained fixing the value of parameter m, the minimum number of contacts considered to compute probability $P(L_{i,j,k}|C^{\geq n}_{i,j,k})$, at 1. We now investigate the impact of varying m on the results, focusing on our best MobHet model, i.e., MobHet-TPC. Considering the two strategies used to define the contacts of a user–*ContactStrength* and *Follower-Followed*, we were able to find users with up to three contacts in the training sets of both datasets. Thus, we experiment with m equal to 1, 2 and 3. The results are shown in Table 5 for the same datasets and contact strategies analyzed in Table 3. Note that, as we increase m, the maximum possible value of θ used by the *ContactStrength* strategy decreases: for $m = 2$, the maximum value of θ is 50%, since this threshold requires that a user must participate in at least half of all interactions (retweets or calls) of u_1 to be u_1's contact. Similarly, a user may have at most one contact when $\theta = 75\%$.

Table 5 shows that the best results are obtained with $m = 1$, implying that, when exploiting the contacts of a user to predict the user's future location, it is better to be less restrictive, and consider even a single contact to perform such inference. The accuracy improvements over stricter policies that consider more contacts ($m = 2, 3$) can be as high as 11%. Moreover, stricter policies may also have a smaller applicability, as there may be fewer users who are candidates for prediction (i.e., users who have at least m contacts). As discussed in Section 3.1, we must revert to a simpler MobHet variation that does not exploit contacts to be able to predict the mobility of such users. As a final note, Table 5 also shows that, for all values of m, the best results are always obtained when the *ContactStrength* strategy with θ equal to 25% or 50% is used to define the contacts, for the reasons discussed above.

5.2. Evaluation of MobHet in other scenarios

As explained in Section 4.2, in addition to using time periods of 60 min with training and testing on the same day with a time window of 5 min, we also evaluate the MobHet models by varying the time window to 10 min and with training and testing on different days.

Table 6 shows the prediction accuracy for the Twitter-BR dataset of Rio de Janeiro on 06/29/2014 and 07/13/2014 as well as the MobilePhone-MX dataset of 03/16/2014 calls and 03/23/2014 in different scenarios. For the scenarios presented in the Table 6, with training and testing on the same day, we use the collection of Rio de Janeiro 06/29/2014 and Mexico 03/16/2014. For results of training and test in different days, we used the dataset of Rio de Janeiro 06/29/2014 for training and 07/13/2014 for testing, and the dataset of Mexico 03/16/2014 for training and 03/23/2014 for testing.

As similar results were obtained for other datasets, we focus our analysis only on the datasets presented in Table 6. Analyzing

Table 5
Impact of parameter m, minimum number of contacts used to compute probability $P(L_{i,j,k}|C^{\geq n}_{i,j,k})$, on MobHet-TPC performance. Best results for each value of m are shown in bold. Overall best results are marked with "*".

Twitter-BR dataset (Rio de Janeiro, 06/29/2014)				
Contact definition		$m = 1$	$m = 2$	$m = 3$
Follower-Followed		0.559 ± 0.0185	0.575 ± 0.0137	0.536 ± 0.0141
	θ = 10%	0.625 ± 0.0195	0.621 ± 0.0132	0.613 ± 0.0132
	θ = 15%	0.640 ± 0.0190	0.621 ± 0.0132	0.613 ± 0.0132
ContactStrength	θ= 20%	0.654 ± 0.0184	0.621 ± 0.0132	0.613 ± 0.0132
	θ = 25%	**0.678 ± 0.0193***	**0.657 ± 0.0135**	**0.624 ± 0.0142**
	θ = 50%	**0.684 ± 0.0197***	**0.645 ± 0.0142**	
	θ = 75%	0.631 ± 0.0192		
MobilePhone-MX dataset				
Contact definition		$m = 1$	$m = 2$	$m = 3$
	θ = 10%	0.630 ± 0.0201	0.621 ± 0.0132	0.613 ± 0.0132
	θ = 15%	0.649 ± 0.0213	0.621 ± 0.0132	0.613 ± 0.0132
ContactStrength	θ = 20%	0.668 ± 0.0210	0.621 ± 0.0132	0.613 ± 0.0132
	θ = 25%	**0.692 ± 0.0172***	**0.657 ± 0.0135**	**0.624 ± 0.0142**
	θ = 50%	**0.683 ± 0.0168***	**0.645 ± 0.014**2	
	θ = 75%	0.618 ± 0.0178		

Table 6
Average precision (along with 95% confidence intervals) of all MobHet models ($m = 1$) for the different scenarios. Best results (and statistical ties) for each method are shown in bold. Overall best results are marked with "*".

Twitter-BR dataset – Rio de Janeiro							
t_k (min)	$\mathcal{D}^{training}$	$\mathcal{D}^{test}$	*MobHet-C (θ = 25%)*	*MobHet-TC (θ = 25%)*	*MobHet-PC (θ = 25%)*	*MobHet-TPC (θ = 25%)*	*MobHet-TP*
5	06/29/2014	06/29/2014	**0.409 ± 0.0149**	**0.620 ± 0.0169**	**0.653 ± 0.0177**	**0.678 ± 0.0193***	**0.545 ± 0.0211**
10	06/29/2014	06/29/2014	0.354 ± 0.0155	0.580 ± 0.0155	0.612 ± 0.0144	0.631 ± 0.0172	0.512 ± 0.0180
5	06/29/2014	07/13/2014	**0.414 ± 0.0152**	**0.625 ± 0.0171**	**0.654 ± 0.0180**	**0.682 ± 0.0180***	**0.555 ± 0.0202**
10	06/29/2014	07/13/2014	0.359 ± 0.0146	0.585 ± 0.0168	0.618 ± 0.0167	0.634 ± 0.0171	0.515 ± 0.0195
MobilePhone-MX dataset – Mexico							
t_k (min)	$\mathcal{D}^{training}$	$\mathcal{D}^{test}$	*MobHet-C(θ = 25%)*	*MobHet-TC(θ = 25%)*	*MobHet-PC(θ = 25%)*	*MobHet-TPC(θ = 25%)*	*MobHet-TP*
5	03/16/2014	03/16/2014	**0.542 ± 0.0230**	**0.643 ± 0.0198**	**0.663 ± 0.0198**	**0.690 ± 0.0172***	**0.618 ± 0.0153**
10	03/16/2014	03/16/2014	0.501 ± 0.0245	0.611 ± 0.0177	0.628 ± 0.0208	0.665 ± 0.0181	0.575 ± 0.0162
5	03/16/2014	03/23/2014	**0.545 ± 0.0220**	**0.645 ± 0.0190**	**0.667 ± 0.0191**	**0.694 ± 0.0170***	**0.619 ± 0.0165**
5	03/16/2014	03/23/2014	0.507 ± 0.0240	0.615 ± 0.0200	0.633 ± 0.0196	0.667 ± 0.0190	0.580 ± 0.0170

Table 7
Average precision (along with 95% confidence intervals) of MobHet-TPC ($m = 1$) and *Null* Model for the different scenarios. Best results (and statistical ties) for each method are shown in bold. Overall best results are marked with "*".

Twitter-BR dataset – Rio de Janeiro				
t_k (min)	$\mathcal{D}^{training}$	$\mathcal{D}^{test}$	*MobHet-TPC (θ = 25%)*	*Null Model*
5	06/29/2014	06/29/2014	**0.678 ± 0.0193***	**0.225 ± 0.0250**
10	06/29/2014	06/29/2014	0.631 ± 0.0172	**0.212 ± 0.0235**
5	06/29/2014	07/13/2014	**0.682 ± 0.0180***	**0.222 ± 0.0248**
10	06/29/2014	07/13/2014	0.634 ± 0.0171	**0.209 ± 0.0253**
MobilePhone-MX dataset – Mexico				
t_k (min)	$\mathcal{D}^{training}$	$\mathcal{D}^{test}$	*MobHet-TPC (θ=25%)*	*Null Model*
5	03/16/2014	03/16/2014	**0.690 ± 0.0172***	**0.242 ± 0.0251**
10	03/16/2014	03/16/2014	0.665 ± 0.0181	**0.235 ± 0.0241**
5	03/16/2014	03/23/2014	**0.694 ± 0.0170***	**0.238 ± 0.0250**
10	03/16/2014	03/23/2014	0.667 ± 0.0190	**0.232 ± 0.0255**

the difference among these scenarios, we observe that scenarios with smaller time windows (5 min) have a prediction accuracy, on average, 4% higher than the scenarios with larger time window (10 min). We observe that with a larger time windows for a small period of simulation time (in our case, 60 min), we could fail to capture some users' transitions between regions and, consequently, compromising the prediction accuraty of MobHet.

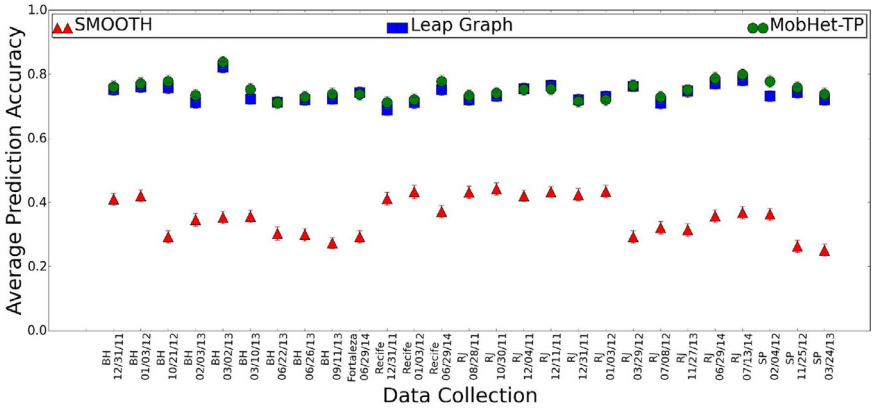
Beside the evaluation for different scenarios, we compared our models with a *Null* model. The Table 7 shows the result of our best model, MobHet-TPC, and the results of the *Null* model for the same scenarios present in Table 6. The MobHet-TPC have, on average, a precision accuracy of 201% higher than the *Null* model, confirming the existence of human mobility in our datasets.

As in our experiments about training and test in different days we did not note no significant difference when compared with the training and test in the same day, we decided to compare our MobHet models with the baselines models, SMOOTH and Leap Graph (in Section 5.3) using the best scenario: training and testing on the same day with the time window of 5 min.

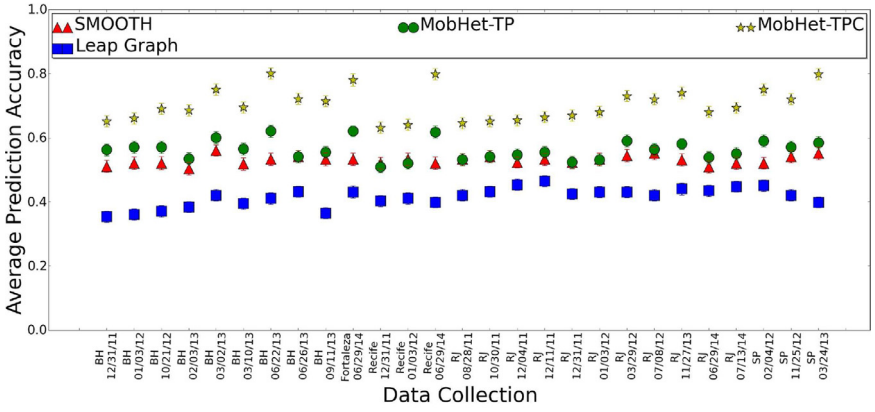
5.3. Comparison of MobHet with baseline models

After comparing our new MobHet models among each other, we now compare them with the two baselines, SMOOTH and Leap Graph. We include in such comparison our best MobHet model, the MobHet-TPC, as well as the MobHet variation that, like the two

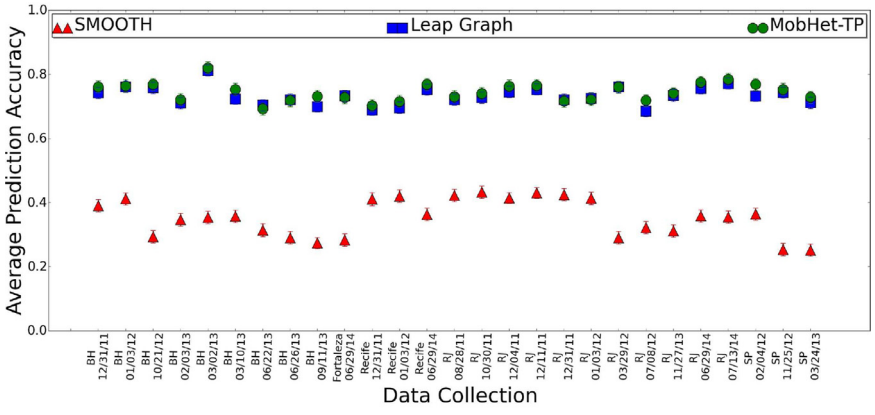
Please cite this article as: L.M. Silveira et al., MobHet: Predicting human mobility using heterogeneous data sources, Computer Communications (2016), http://dx.doi.org/10.1016/j.comcom.2016.04.013



(a) Only calls as input (MobilePhone-BR dataset)



(b) Only tweets as input (Twitter-BR dataset)



(c) Both calls and tweets, associating tweets to calls (both datasets)

Fig. 7. Average prediction accuracy of our best MobHet models and baselines on all Brazilian datasets (Twitter-BR and MobilePhone-BR)—Training and test on the same day and time window of 5 min.

baselines, does *not* exploit user contacts, the MobHet-TP. For the former, we set $m = 1$ and focus on the contact definition that produces the best results, that is, *ContactStrength* with θ equal to 25% and 50%.

Table 8 shows average prediction accuracy (along with 95% confidence intervals) for all models and the three datasets, considering scenarios in which only calls are used (MobilePhone-BR and MobilePhone-MX datasets), only tweets are used (Twitter-BR dataset) as well as both types of data are used (MobilePhone-BR and Twitter-BR datasets). For the latter, we consider both approaches to combine the data discussed in Section 4.2: association of *tweets to calls* and association of *calls to tweets*. For both MobilePhone-BR and Twitter-BR datasets, we present results for a single city and period to improve the readability of the table. We will discuss the results for the other cities and periods, which are very similar, later in this section.

Concerning the baselines, we note that both SMOOTH and Leap Graph produce the best results when using the homogeneous data source for which the model was originally proposed (or evaluated). That is, Leap Graph achieves its best prediction accuracy when using only phone calls, while SMOOTH's best results are obtained when using only tweets. Specifically, by looking at the results

Table 8
Comparison of best MobHet models against baselines: average prediction accuracy and 95% confidence interval. Best results for each scenario in bold.

MobilePhone-BR and Twitter-BR datasets				
Rio de Janeiro, 06/29/2014				
Models	Calls	Tweets	Tweets to calls	Calls to tweets
SMOOTH	0.368 ± 0.0177	0.521 ± 0.0189	0.481 ± 0.0193	**0.516 ± 0.0189**
Leap Graph	**0.788 ± 0.0178**	0.451 ± 0.0195	**0.745 ± 0.0187**	0.422 ± 0.0189
MobHet-TP	**0.799 ± 0.0192**	0.545 ± 0.0211	**0.744 ± 0.0186**	**0.531 ± 0.0164**
MobHet-TPC ($\theta = 25\%$)		**0.678 ± 0.0193**		
MobHet-TPC ($\theta = 50\%$)		**0.684 ± 0.0197**		
MobilePhone-MX dataset				
Models	Calls	Tweets	Tweets to calls	Calls to tweets
SMOOTH	0.345 ± 0.0221			
Leap Graph	0.613 ± 0.0199			
MobHet-TP	0.619 ± 0.0153			
MobHet-TPC ($\theta = 25\%$)	**0.692 ± 0.0172**			
MobHet-TPC ($\theta = 50\%$)	**0.683 ± 0.0168**			

produced for the Brazilian datasets (MobilePhone-BR and Twitter-BR), we note a performance degradation of 43% for Leap Graph and 29% for SMOOTH if a different type of data (though still homogeneous) is used. As for the scenarios with heterogeneous data sources, the prediction accuracy of each model was somewhat worse that its best results. Thus, we observe that the performance of both baseline is very dependent on the type of data used as input, and suffers some degradation when heterogeneous data sources are used as input.

In contrast, MobHet-TP, our model that, like the two baselines does *not* exploit user contacts, has a performance that is at least as good as (if not better than) that of the best baseline in *all* scenarios. For example, we note that MobHet-TP has an average prediction accuracy that is slightly better than that of Leap Graph, but much higher (117% in MobilePhone-BR and 80% in MobilePhone-MX datasets) than SMOOTH in the scenarios with only calls as data source. If only tweets are used as input data, MobHet-TP outperforms Leap Graph by 21%, producing results similar to those of SMOOTH (Twitter-BR dataset). Note that MobHet-TP results are much better when only calls as used, compared to when tweets are used. The reason for that is a much larger number of distinct regions which are present only in the test set (but not in the training set) of the Twitter-BR dataset. The model is not able to predict a movement towards a region that did not appear in the training set, since this region is not included in transition graph G.

In the scenario with heterogeneous data sources (Brazilian datasets), MobHet-TP features a performance slightly worse when compared with the scenarios with a single homogeneous data source, but still higher on average than that of the baselines. As to the two strategies to combine the data sources, we note that all models perform between when the association targets the type of data for which the model has higher accuracy (e.g., calls for Leap Graph and MobHet-TP, tweets for SMOOTH).

The introduction of user contacts into MobHet produces further improvements over both baselines. For example, in the Twitter-BR dataset, our best MobHet-TPC model outperforms SMOOTH by 31% and Leap Graph by 52%. In the MobilePhone-MX dataset, the improvements are 90% and 13% respectively. Indeed, by cross-referencing Table 3 and 8, we note that even when the less effective contact definition (*Follower-Followed* on Twitter-BR, and *ContactStrength* with $\theta = 75\%$ on MobilePhone-MX) already produces improvements over both baselines.

Although the results shown in Table 8 for the Brazilian datasets are for a single city and time period, the same relative performance of all methods was observed for all 27 individual collections that compose those datasets (see Table 2). This is illustrated in Fig. 7(a–c) for three scenarios, two with homogeneous data and one with heterogeneous data sources and with time window of 5 min and training and test at the same day. Overall, when using only calls as input data (Fig. 7(a)), MobHet-TP produces accuracy improvements of 6% and 114% over Leap Graph and SMOOTH, respectively, on average. When using only tweets as input (Fig. 7(b)), our best model—MobHet-TPC with $\theta = 50\%$ produce average improvements of 71% and 33%, respectively. In the scenario with heterogeneous data (Fig. 7(c)), the improvements are still quite noticeable, reaching, on average, 110% over SMOOTH, and 4% over Leap Graph.

6. Conclusion and future work

We have proposed MobHet, a new family of models to predict human mobility from heterogeneous data sources. The MobHet models exploit a subset of the popularity of different regions in a target area, the frequency at which people moves between different regions as well as the relationships (or contacts) among people. We evaluated our proposed models, comparing them with themselves and with two baseline solutions from the literature, in various scenarios, with homogeneous and heterogeneous data, built from large real-world datasets of mobile phone calls and tweets. Our experiments indicate that neither baseline outperforms the other in all scenarios, demonstrating their sensitivity to the type of input data. In contrast, our MobHet models are at least as good as, if not much better than, the best baseline in all scenarios. Moreover, for all scenarios with a varied set of parameters, we find that all three features—region popularity, frequency of transition between regions and user contacts—are important to mobility prediction, since leaving any of them out causes loss of prediction accuracy. Regarding specifically the definition of user contacts, our results show that less restrictive strategies may lead to very unreliable contact inferences, ultimately hurting prediction. On the other extreme, very strict contact inferences may excessively constraint the size of the training set, which in turn also hurts model generality and accuracy.

This work opens up many perspectives for future work building upon our current models. For example, we intend to further investigate other alternatives to define user contacts, possibly exploiting temporal and spatial information. In that direction, one could envision different *classes* of contacts, such as those that a user often meet at daytime and those whose interaction occur more ofter at nighttime. Another direction we plan to pursue in the future relates to the division of the target area into a set of regions. Instead of taking a uniform division (as performed here), we intend to explore approaches that take sociocultural, demographic, and/or

Please cite this article as: L.M. Silveira et al., MobHet: Predicting human mobility using heterogeneous data sources, Computer Communications (2016), http://dx.doi.org/10.1016/j.comcom.2016.04.013

administrative aspects into account. For example, one alternative approach would be to break the target area of a city into city districts. Another example is to use other features connected to the user, like his/her historical record to predict the location. Mobility prediction in such scenario could provide valuable insights into more effective policies for city planning. Finally, we also intend to explore other mobility prediction tasks. For example, we intend to develop models to predict the *volume* of people who will be at a certain region in a target time period.

Acknowledgements

This work was supported by FAPEMIG (FAPEMIG-PRONEX-MASWeb, Models, Algorithms and Systems for the Web, process number APQ-01400-14), FAPERJ, CAPES, INWEB and CNPq. In particular, we acknowledge the STIC-AmSud Program that supported meetings that enabled this work.

References

[1] M.C. González, C.A. Hidalgo, A.-L. Barabási, Understanding individual human mobility patterns., Nature 453 (7196) (2008).
[2] D. Balcan, V. Colizza, B. Gonçalves, H. Hu, J.J. Ramasco, A. Vespignani, Multiscale mobility networks and the spatial spreading of infectious diseases, Proceedings of the National Academy of Sciences (PNAS) 106 (51) (2009).
[3] F.H.Z. Xavier, L.M. Silveira, J.M. Almeida, A. Ziviani, C.H.S. Malab, H.T. Marques-Neto, Analyzing the workload dynamics of a mobile phone network in large scale events, in: Proceedings of the Workshop on Urban Networking (UrbaNe), ACM CoNEXT, 2012.
[4] H. Gao, J. Tang, H. Liu, Addressing the cold-start problem in location recommendation using geo-social correlations, Data Min. Knowl. Discov. 29 (2) (2014).
[5] J. Bao, Y. Zheng, D. Wilkie, M. Mokbel, Recommendations in location-based social networks: a survey, GeoInformatica 19 (3) (2015).
[6] X. Xiao, Y. Zheng, Q. Luo, X. Xie, Inferring social ties between users with human location history, J. Ambient Intell. Humaniz. Comput. 5 (1) (2014).
[7] C. Song, Z. Qu, N. Blumm, A.-L. Barabási, Limits of predictability in human mobility, Science 327 (5968) (2010).
[8] W. Dong, N. Duffield, Z. Ge, S. Lee, J. Pang, Modeling cellular user mobility using a leap graph, in: Proceedings of the International Conference on Passive and Active Measurement (PAM), 2013.
[9] A. Munjal, T. Camp, W.C. Navidi, SMOOTH: a simple way to model human mobility, in: Proceedings of the ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM), 2011.
[10] K. Lee, S. Hong, S.J. Kim, I. Rhee, S. Chong, SLAW: self-similar least-action human walk, IEEE/ACM Trans. Netw. 20 (2) (2012).
[11] M. Allamanis, S. Scellato, C. Mascolo, Evolution of a location-based online social network: Analysis and models, in: Proc. of the ACM Internet Measurement Conference (IMC), 2012.
[12] A. Hess, K.A. Hummel, W.N. Gansterer, G. Haring, Data-driven human mobility modeling: A survey and engineering guidance for mobile networking, ACM Comput. Surv. 48 (3) (2015).
[13] S. Scellato, A. Noulas, R. Lambiotte, C. Mascolo, Socio-spatial properties of online location-based social networks., in: Proceedings of the AAAI Int. Conf. on Weblogs and Social Media (ICWSM), 2011.
[14] E. Cho, S.A. Myers, J. Leskovec, Friendship and mobility: user movement in location-based social networks, in: Proceedings of the ACM International Conference on Knowledge Discovery and Data Mining (SIGKDD), 2011.
[15] N. Li, G. Chen, Multi-layered friendship modeling for location-based mobile social networks, in: Proceedings of the Conferece on Mobile and Ubiquitous Systems: Networking Services (MobiQuitous), 2009.
[16] D. Wang, C. Song, Impact of human mobility on social networks, J. Commun. Netw. 17 (2) (2015).
[17] N. Bui, N. Bui, F. Michelinakis, F. Michelinakis, J. Widmer, A model for throughput prediction for mobile users, in: Proceedings of the European Wireless Conference, 2014.
[18] M. Li, A. Ahmed, A.J. Smola, Inferring movement trajectories from GPS snippets, in: Proceedings of the ACM International Conference on Web Search and Data Mining (WSDM), 2015.
[19] I. Rhee, M. Shin, S. Hong, K. Lee, S. Chong, On the Levy-walk nature of human mobility, Proceedinds of the INFOCOM, 2008.
[20] A.I.J. Tostes, F. de L. P. Duarte-Figueiredo, R. Assunção, J. Salles, A.A.F. Loureiro, From data to knowledge: city-wide traffic flows analysis and prediction using bing maps, in: Proceedings of the ACM SIGKDD International Workshop on Urban Computing (UrbComp), 2013.
[21] A. Noulas, S. Scellato, N. Lathia, C. Mascolo, A random walk around the city: New venue recommendation in location-based social networks, in: Proceedings of the International Conference on Social Computing (SocialCom), 2012.
[22] A. Noulas, C. Mascolo, Exploiting foursquare and cellular data to infer user activity in urban environments, in: Proceedings of the IEEE International Conference on Mobile Data Management (MDM), 2013.
[23] C.A. Davis Jr., G.L. Pappa, D.R.R. de Oliveira, F. de L. Arcanjo, Inferring the location of twitter messages based on user relationships, Trans. in GIS 15 (6) (2011).
[24] T. Nguyen, B.K. Szymanski, Using location-based social networks to validate human mobility and relationships models, in: Proceedings of the International Conference on Advances in Social Networks Analysis and Mining (ASONAM), 2012.
[25] B. Alharbi, X. Zhang, Exploring the significance of human mobility patterns in social link prediction, in: Proceedings of the ACM Symposium on Applied Computing (SAC), 2014.
[26] J. Candia, M.C. González, P. Wang, T. Schoenharl, G. Madey, A.-L. Barabási, Uncovering individual and collective human dynamics from mobile phone records, J. Phys. A: Math. Theor. 41 (22) (2008).
[27] K. Lee, S. Hong, S.J. Kim, I. Rhee, S. Chong, SLAW: a new mobility model for human walks, in: Proceedings of the INFOCOM, 2009.
[28] M. Musolesi, C. Mascolo, Designing mobility models based on social network theory, ACM SIGMOBILE Mobile Computing and Communications Review 11 (3) (2007).
[29] A. Noulas, S. Scellato, R. Lambiotte, M. Pontil, C. Mascolo, A tale of many cities: universal patterns in human urban mobility, PloS ONE 7 (5) (2012).
[30] L.M. Silveira, J.M. Almeida, H.T. Marques-Neto, A. Ziviani, MobDatU: a new model for human mobility prediction based on heterogeneous data, in: Proceedings of the Brazilian Symposium on Computer Networks and Distributed Systems (SBRC), 2015.