There's A Path For Everyone: A Data-Driven Personal Model Reproducing Mobility Agendas

Riccardo Guidotti*, Roberto Trasarti*, Mirco Nanni*, Fosca Giannotti*, Dino Pedreschi§
* KDD Lab, ISTI - CNR, Via Giuseppe Moruzzi, 1, Pisa, Italy, {name.surname}@isti.cnr.it
§ KDD Lab, University of Pisa, Largo B. Pontecorvo, 3, Pisa, Italy, {name.surname}@di.unipi.it

***Abstract*—The avalanche of mobility data like GPS and GSM daily produced by each user through mobile devices enables personalized mobility-services improving everyday life. The base for these mobility-services lies in the predictability of human behavior. In this paper we propose an approach for reproducing the user's personal mobility agenda that is able to predict the user's positions for the whole day. We reproduce the agenda by exploiting a data-driven personal mobility model able to capture and summarize different aspects of the systematic mobility behavior of a user. We show how the proposed approach outperforms typical methodologies adopted in the literature on four different real GPS datasets. Moreover, we analyze some features of the mobility models and we discuss how they can be employed as agents of a simulator for what-if mobility analysis.**

I. Introduction

Nowadays we have the unprecedented opportunity of closely observe human mobility through data analysis. Each one of us produces an enormous amount of mobility data while performing our daily activities: the massive use of digital systems and mobile devices leaves behind a myriad of *digital traces* like GPS and Call Data Records data originated by the usage of our smartphones. Telephone calls, SMS, Google Location History, Google Maps, Waze, geo-referenced Tweets and Facebook posts are few examples of services producing these data. Moreover, many insurance companies install on the insured cars black-boxes storing the GPS positions.

These digital traces take note of our mobility behavior with extraordinary precision and can be exploited for realizing the most disparate location-based services [1]: recommender systems [2], [3], personalized journey planners [4], [5], car-pooling systems [6], [7], etc. Such services are based on the *predictability* of human behavior: typically every individual systematically repeats a small set of actions [8] such as visiting a limited number of places [9]. Thus, the first step towards a good mobility service is a mobility *predictor*. Predicting the destination location [10], or the movement along a trajectory [11] is a problem already addressed in literature with good results. However, as will be discussed in Sec. III, predicting the *mobility agenda* containing all the positions of the user for the whole day is a much more complex problem.

In this paper we propose an approach for *extracting* the user's *personal mobility model* and then we use it to *reproduce* the user's *personal mobility agenda* representing the predicted positions where the user accomplish her activities during the whole day. The approach we propose is called *RAMA* – *Routinary Actions Mobility Agenda* – and it is *completely unsupervised* and adaptive to different users and mobility scenarios. These properties are provided to *RAMA* by the model defined in Sec. IV which is designed by combining in an innovative way the best features of several existing state-of-the-art mobility models. In [12] the *mobility profile* and *routines* are defined, which are then used in [11] to build a *trajectory predictor*. In [13] the mobility model is represented as a *network* with specific properties. In [14] it is shown the advantage of *adaptive unsupervised algorithm* in the discovery of *user's locations*. The proper fusion of these features leads to a novel model that is a more complex, complete and representative solution for capturing personal mobility.

In the experimental section (Sec. V) we show how *RAMA* outperforms the naive predictors and the typical ones used in the literature. Then, we analyze how *RAMA* can be instantiated in different ways varying the time extent of the agenda. Moreover, with respect to the reproduction task, we also consider the real time reproduction of the agenda taking advantage of the information coming from the user and correcting the agenda if something changes or is not as expected. Empirical results highlight how flexible and adaptable *RAMA* is on real datasets of private users in *Rome*, *London*, *Boston* and *Beijing*. Furthermore, we emphasize also some analytical properties of the proposed mobility model for the cities analyzed.

We conclude the paper (Sec. VI) discussing how the agenda reproduction is only one of the possible examples of how the personal mobility model proposed can be used in a real application scenario. Indeed, besides *prediction*, the objective of building a model able to capture and reproduce user's mobility is a major requirement of *simulation* applications. The difference is that in *prediction* the purpose of the models is to recognize a *match* between the model and the current behavior of a user for predicting her future mobility, while in *simulation* the objective of the models is to be integrated as *agents* into existing simulators.

II. Related Work

The challenge of defining a user's mobility model is commonly addressed in literature using three approaches: Markov chains, mixture of general laws and pattern discovery.

An example belonging to the first category is [15] which characterizes and classifies user's Point of Interests (POIs) according to their relevance for the user: *mostly visited* POIs, *occasionally* visited POIs, *exceptionally* visited POIs, and build a Markov chain using the movements and the stops

duration to weight the chain. In [16] the authors uses Markov chains and a mixture of data-driven mobility laws to generalize the user's behavior from the geography, and to describe them in terms of their preferential exploration or return tendencies [17]. Then they use the models to generate synthetic trajectories maintaining some properties of original data, e.g. number of locations per user, radius of gyration, mobility entropy.

In [18] a non-parametric Bayesian method for modeling collections of timestamped events is proposed. The authors use a Dirichlet process for learning a set of intensity functions which form a basis set for representing individual time-periods which are exploited for "unusual" events detection. A general law approach is also followed by [8] where the authors state that human trajectories show a high degree of spatio-temporal regularity, each individual being characterized by a time-independent characteristic travel distance and a significant probability to return to a few highly frequented locations.

The work in [19] presents a data-driven approach which uses mobility patterns. Those patterns are used to predict the future positions of a user when it matches the pattern premises. Other approaches related to pattern discovery consider also external factors such as social relationship. In [20] a locations recommender is presented, based on past user behavior, the locations' venues, the social relationships and the similarity among users. In [21] it is empirically observed that the movements of a user can be classified as *short* and *long* distance travels, where the former is based on periodic behaviors, while the latter is more likely to happen due to social influence.

Besides the type of model, another crucial aspect is the strategy adopted for managing the spatial and temporal dimensions. Most of the existing approaches discretize them using grids computed over the whole data. In [22] the authors map the real position of the users in an hexagonal grid. [23] makes use of a grid-based technique to extract interesting locations. This over-generalization of the locations simplifies the problem and makes it more manageable. Doing it individually for each user is more difficult and requires finer-granularity approaches but, as we will show, the performance greatly increase.

The model we propose differs from those described above in different aspects: *(i)* our approach is *user-centric*, no global knowledge is pushed from the expert standardizing the users in any aspect (e.g. discretization of space, number of locations, etc.); *(ii)* it is fully *automatic and adaptive*, no parameters are used by the analyst to drive the discovery of regularity (e.g. frequency thresholds); and *(iii)* it considers the user's behavior as a *continuous flow* of decisions taken in different contexts. We show how those core principles shape the analytical process and their effectiveness in real case scenarios.

III. Problem Formulation

The problem we face consists in approximating the personal mobility agenda that each specific user of a given population is going to follow during a day in the near future. The agenda is informally defined as the sequence of the positions in which she is going to be, instant by instant, during the whole day.

Before providing a more rigorous definition of agendas we introduce two basic concepts: *trajectory*, which is the basic input data type; and *personal mobility history*, representing the trips traveled by a user in a specific period [24].

Definition 1 (Trajectory)**.** *A* trajectory *is a sequence* $t{=}\langle p_1, \ldots, p_n \rangle$ *of spatio-temporal points, each being a tuple* $p_i = (lon_i, lat_i, ts_i)$ *that contains longitude* lon_i*, latitude* lat_i *and timestamp* ts_i *of the point. The points of a trajectory are chronologically ordered, i.e.,* $\forall\ 1 \leq i < n\ :\ ts_i < ts_{i+1}$.

Given a trajectory t we refer to its i-th point p_i with the notation $t[i]$, and to its number of points with $t.n$. Also, we indicate the longitude, latitude and timestamp components of point $t[i]$ respectively with the notation $t[i].lon$, $t[i].lat$, and $t[i].ts$. For a timestamp ts we indicate its associated date and time of the day with $date(ts)$ and $time(ts)$ respectively.

Definition 2 (History)**.** *The* history $H_u^{d,d'}$ *of a user* u *is the set of trajectories traveled between dates* d *and* d'*:* $\forall\ t \in H_u^{d,d'}$: $d \leq date(t[1].ts) \leq date(t[t.n].ts) \leq d'$. $H_u^{d,d'}$ *is denoted* H_u^d *when* $d = d'$*, and* H_u *when* $d = -\infty$ *and* $d' = \infty$.

We can then define the *personal mobility agenda* of a user, which basically provides the position of the user throughout a day at a constant sampling rate.

Definition 3 (Agenda)**.** *Given the history* H_u^d *of user* u *in date* d *and a time granularity* $\tau \in R^+$ *(also called clock tick), we define the corresponding* user's agenda $A_u^d(\tau)$ *as a trajectory that covers the whole day at fixed sampling rate* τ*, i.e.* $A_u^d(\tau) = t$ *with* $t.n = \lfloor 24h/\tau \rfloor$ *and* $\forall\ 1 \leq i \leq t.n$: $date(t[i].ts) = d \wedge time(t[i].ts) = \tau * (i-1)$. *Latitude and longitude of each point* $p \in A_u^d(\tau)$ *is determined as follows: if* $\exists t \in H_u^d : t[1].ts \leq p.ts \leq t[t.n].ts$*, then they are derived through linear interpolation from* t*; otherwise they take the same position of the temporally closest point in all* H_u^d.

The objective of agendas is to make the information about the user's position provided by her history more complete and uniform in time. That is required for a few reasons. First, in practice, the trajectories of a user's history describe her position only during movement, while the periods of stop are missing – or, more exactly, implicit. That holds in particular for the time periods that precede the first trajectory of the day or that follow the last one, which are typically significantly large; second, the points contained in H_u^d are usually temporally irregular, which makes it difficult to work with them. $A_u^d(\tau)$ overcomes the problem by modeling the user's positions at fixed times according to the clock tick τ, which is realized computing the (expected) position at each instant through a linear interpolation. We assume that between two consecutive points of a trajectory each user follows a uniform, linear movement with constant direction and speed.

As an example, let consider the trajectories in $H_u^d{=}\{t_1, t_2\}$:

$t_1{=}\langle$(12.489, 41.922, 08:12), (12.478, 41.928, 08:20), (12.466, 41.928, 08:27)$\rangle$

$t_2{=}\langle$(12.466, 41.928, 16:34), (12.473, 41.919, 16:41), (12.488, 41.921, 16:57)$\rangle$

The corresponding agenda with clock tick $\tau = 10min$ is the following, composed by $24h/10min = 144$ points:

$$A_u^d(10min) = \langle (12.489, 41.922, 00{:}00), (12.489, 41.922, 00{:}10), \ldots,$$
$$(12.489, 41.922, 08{:}10), (12.478, 41.928, 08{:}20),$$
$$(12.466, 41.928, 08{:}30), (12.466, 41.928, 08{:}40), \ldots,$$
$$(12.466, 41.928, 16{:}30), (12.473, 41.918, 16{:}40),$$
$$(12.486, 41.919, 16{:}50), (12.488, 41.921, 17{:}00), \ldots,$$
$$(12.488, 41.921, 23{:}40), (12.488, 41.921, 23{:}50)\rangle$$

The colors highlight the dependencies between points in the agenda and the corresponding points that were used to infer them. In particular, the points in $A_u^d(10min)$ corresponding to no movement are simple duplicates of points in H_u^d (e.g. times 0:00, 0:10, etc.), whereas some points during movement are obtained through interpolation in correspondence of clock ticks (e.g. time 16:50). Notice that, as usually happens also in real data, the ending point of t_1 and the starting of t_2 coincide.

Finally, the problem we tackle in this paper is the following:

Definition 4 (Agenda Reproduction Problem)**.** *Given the history $H_u^{d',d''}$ of user u between dates d' and d'', a clock tick τ and a later date $d > d''$, the* agenda reproduction problem *consists in inferring an approximate agenda $\mathbb{A}_u^d(\tau)$ for date d, called* reproduced agenda*, that is as close as possible to the real one w.r.t. a comparison function $agendaDist$:*

$$agendaDist(A_u^d(\tau), \mathbb{A}_u^d(\tau)) \approx 0$$

There are various challenges related to this problem. First, differently from most common formulations, such as location [10] or trajectory [11] prediction, here we have no knowledge about the current or past positions of the user for the prediction date. Second, another difficult issue which is rarely addressed in the literature on mobility prediction/simulation is the estimation of the duration of the stop that the user performs on a location before starting a new movement. Third, producing agendas containing fine-grained, GPS-like information increases the complexity of the task, making it more difficult than working with pre-defined areas [11]. Finally, the possible reproduced agendas are strongly dependent on the area where the user lives and moves, with its mobility and traffic.

IV. Proposed Approach

We solve the agenda reproduction problem by proposing a two-step approach: *(i)* we learn a personal mobility model by observing the personal mobility history, and *(ii)* we exploit the model learned to reproduce future personal mobility agendas.

Most of the approaches in the literature related to modeling mobility behaviors suffer from various weaknesses. Indeed, in order to reduce the complexity of the problem generated by accurate GPS data, a very common procedure consists of employing forms of spatio-temporal discretization like a simple spatio-temporal grid. On one hand, this makes easier to find frequent or interesting areas and mobility patterns. On the other hand, it affects the precision of the applications they are aimed for, since they can only infer areas with a granularity imposed by the apriori discretization. This weakness sometimes is overtaken by adopting smarter forms of spatial discretization, like clustering algorithms. Most of these algorithms require a parameter setting (e.g. the radius to decide when two points should belong to the same cluster, etc.) that is generally imposed to be equal for all the users. Unfortunately, it has been extensively proved that such general settings are frequently incorrect for the personal data of an individual user [24] and might cause an algorithm to fail in finding the true patterns, or make the algorithm report patterns that do not really exist [25]. Finally, also temporal discretization precludes a model from considering the time continuity.

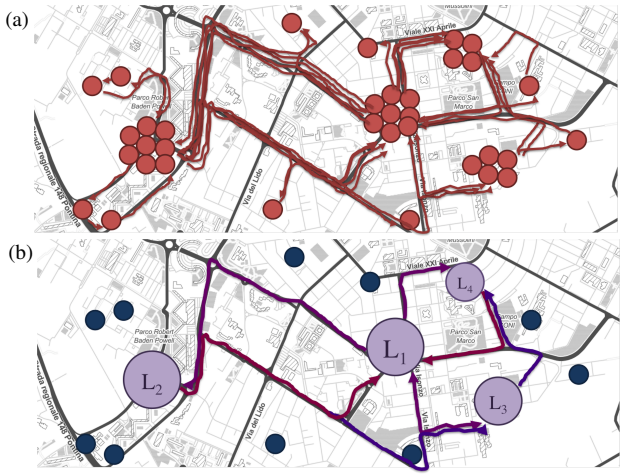

Fig. 1: *(a)* History H_u: u followed 34 trajectories (red arrows) with 35 stops (red circles), *(b)* Model P_u summarizes H_u with 4 regular locations (big purple circles) and 8 movements (violet arrows).

A. Building Personal Mobility Models

In this section we define the personal mobility model exploited by the agenda reproduction approach that we propose. The model is able to capture the systematic presences of the user in her most frequent locations, and the routinary movements that lead the user from a location to another one. Moreover, in order not to suffer from the weaknesses mentioned above, the personal mobility model is built *(i)* without requiring any apriori spatial or temporal discretization, *(ii)* in an auto-adaptive fashion and without the need of any form of parameter tuning for different users, *(iii)* keeping time granularity sufficiently fine to approximate continuity.

Given a user u and her *history* $H_u = \{t_1, \ldots, t_n\}$ (see Fig. 1 *(a)* for an example), in the following of this section we define the components of the personal mobility model P_u.

Definition 5 (Stops)**.** *Given the history H_u of user u, we define the* stops *of u as $S_u = \bigcup_{t \in H_u} \{t[1], t[t.n]\}$.*

We name $getStops(H_u) = S_u$ the function that takes as input the mobility history and returns the set of stops.

As can be observed in Fig. 1 *(a)*, even though these stops represent GPS-like locations, and therefore highly variable positions, they can be conceptually separated into homogeneous groups, each defining a particular place (or area) they cover. For example, it can be the case of home or work, and their associated stops might correspond to parking lots in the surroundings (see the two big groups of points in Fig. 1 *(a)*).

Definition 6 (Locations)**.** *Given the stops S_u of user u, we define* locations $\mathcal{L}_u = \{L_1, \ldots, L_k\}$ *as a partitioning of S_u into disjoint sets of similar stops.*

In our example the locations are the purple and blue circles in Fig. 1 *(b)*. We name $getLocations(S_u) = \mathcal{L}_u$ the function that takes the user's stops and returns the locations.

Besides locations, the mobility of a user is characterized by *movements*, i.e., trajectories with a similar purpose.

Definition 7 (Movements)**.** *Given the history H_u and locations $\mathcal{L}_u$ of user u, we define her* movements $\mathcal{M}_u = \{M_1, \ldots, M_m\}$ *as a partitioning of H_u into disjoint sets such that $\forall M \in \mathcal{M}_u : \exists L, L' \in \mathcal{L}_u$ s.t. $M = \{t \in H_u | t[1] \in L \wedge t[n] \in L'\}$. We denote with $\overline{\mathcal{M}}_u$ the corresponding set of links among locations $\overline{\mathcal{M}}_u = \{(L, L') | L, L' \in \mathcal{L}_u \wedge \exists t \in H_u . t[1] \in L \wedge t[n] \in L'\}$.*

In other words, a movement is a set of trajectories which start from a location L and end in a location L'. Each trajectory belongs only to a movement. We define with $getMovements(H_u, \mathcal{L}_u) = \mathcal{M}_u$ the function that takes the history and the locations and returns the set of movements.

The locations and movements can be thought conceptually combined together using a network data structure which links the elements in a natural way from a mobility point of view, generating the so called *personal mobility network* [13]:

Definition 8 (Mobility Network)**.** *Given the locations $\mathcal{L}_u$ and movements $\mathcal{M}_u$ of a user u, we define her* mobility network *as a directed graph $G_u = (\mathcal{L}_u, \overline{\mathcal{M}}_u)$, where $\mathcal{L}_u$ is the set of nodes and $\overline{\mathcal{M}}_u$ is the set of edges. On the nodes and edges we define the following behavioural functions:*

- $\omega : \mathcal{L}_u \to \mathcal{N}$ *returns the number of times u stopped in a location $L \in \mathcal{L}_u$, i.e., $|\{t \in H_u | t[t.n] \in L\}|$.*
- $\rho : \mathcal{L}_u \times Time \to [0, 1]$ *estimates the probability to find u in a location $L \in \mathcal{L}_u$ at time $ts \in Time$.*
- $\pi : \mathcal{L}_u \times \mathcal{L}_u \times Time \to [0, 1]$ *estimates the probability that user u at time ts moves from L to L' ($L, L' \in \mathcal{L}_u$).*

Assuming that H_u is complete, i.e., the data contains no gaps, then G_u *has no terminal node*, since any trip arriving in a location is followed by another leaving it, and therefore, in a simulation perspective, any agent randomly moving according to G_u should never get stuck in a location. The only exception is when the last location visited in the history is a new one, yet that seems to be an unlikely event, and indeed we empirically observed that this never happens for the users analyzed.

The model that we are defining might suffer from the noise generated by occasionally visited places and routes which are not habitual for the user. To mitigate the problem, we filter out the infrequent locations and movements by focusing on the set of *regular locations* of each user u: $\mathcal{L}_u^{\Re} \subseteq \mathcal{L}_u$. We name $getRegularLocations(\mathcal{L}_u) = \mathcal{L}_u^{\Re}$ the function that takes the locations and returns the regular locations. Accordingly, all the movements starting or ending in an occasional location, i.e., $L \notin \mathcal{L}_u^{\Re}$, are removed, which however might lead to have nodes with no outgoing edges. Since that is detrimental to usability for reproduction purposes, we iteratively remove all the existing terminal nodes and their respective movements according to G_u, stopping when no one is left or, in the extreme case, only two nodes remain. We empirically verified that the latter case never occurs in our data: all the users remained with at least five nodes. We indicate with $\mathcal{M}_u^{\Re}$ the set of *regular movements* passing the filtering phase. All behavioral functions of G_u are adjusted after this filtering phase by updating the probabilities. Notice that the resulting mobility network with the behavioral functions recalls a Markov chain model [26] which is frequently used in simulation.

Since agendas prescribe a precise position for each clock tick while our model groups points and trajectories, we need to associate a representative to each location and movement.

Definition 9 (Location-Point)**.** *Given a location L and a comparison function among points $pointDist$, we define* location-point *as the point l such that*

$$l = \arg\min_{p \in L} \sum_{p' \in L, p \neq p'} pointDist(p, p')$$

The location-point l is the best point representing all the stops in a location L. We name $getLocationsPoints(\mathcal{L}_u) = Q_u$ the function that takes the locations and returns the locations-points, and $Q_u^{\Re}$ the set of regular location-points.

Definition 10 (Routine)**.** *Given a movement $\mathcal{M}$ and a comparison function among trajectories $trajDist$, we define its representative* routine *as the trajectory r such that*

$$r = \arg\min_{t \in M} \sum_{t' \in M, t \neq t'} trajDist(t, t')$$

We name $getRoutines(\mathcal{M}_u) = R_u$ the function that takes the movements and returns the routines. We indicate with $R_u^{\Re}$ the routines associated to regular movements $\mathcal{M}_u^{\Re}$. The violet and purple arrows in Fig. 1 *(b)* represent the routines.

Regular locations, regular movements and the adjusted behavioral functions describe the personal mobility model:

Definition 11 (Personal Mobility Model)**.** *Given the regular locations $\mathcal{L}_u^{\Re}$, location-points $Q_u^{\Re}$, movements $\mathcal{M}_u^{\Re}$ and routines $R_u^{\Re}$ of user u, we define the* personal mobility model $P_u = (G^{\Re}, Q_u^{\Re}, R_u^{\Re}, \omega, \rho, \pi)$ *where $G^{\Re} = (\mathcal{L}_u^{\Re}, \overline{\mathcal{M}}_u^{\Re})$ is the regular graph and ω, ρ, π are the behavioral functions relative to the regular locations and movements.*

The personal mobility model P_u accurately describes and summarizes the user's mobility habits without any spatial or temporal discretization, or parameter setting. Fig. 1 *(b)* depicts the regular graph of a sample personal mobility model.

B. Realizing Personal Mobility Models

In this section we summarize the workflow for extracting the model P_u from the history H_u and how we realize the functions defined in the previous section.

Alg. 1 summarizes the required steps. The first one is a selection (line 1) of the stops from input trajectories. We realize the stops partitioning (line 2) using TOSCA [14], a parameter-free clustering algorithm designed for personal

Algorithm 1: $modelExtraction(H_u)$

1 $S_u \leftarrow getStops(H_u)$;
2 $\mathcal{L}_u \leftarrow getLocations(H_u)$;
3 $\mathcal{M}_u \leftarrow getMovements(H_u, \mathcal{L}_u)$;
4 $G_u \leftarrow (\mathcal{L}_u, \mathcal{M}_u)$;
5 $\mathcal{L}_u^{\Re} \leftarrow getRegularLocations(\mathcal{L}_u)$;
6 $\mathcal{L}_u^{\Re}, \mathcal{M}_u^{\Re} \leftarrow filterOccasional(G_u, \mathcal{L}_u^{\Re})$;
7 $Q_u^{\Re} \leftarrow getLocationsPoints(\mathcal{L}_u^{\Re})$;
8 $R_u^{\Re} \leftarrow getRoutines(\mathcal{M}_u^{\Re})$;
9 $G_u^{\Re} \leftarrow (\mathcal{L}_u^{\Re}, \mathcal{M}_u^{\Re})$;
10 $\omega, \rho, \pi \leftarrow getBehavioralFunctions(G_u^{\Re})$;
11 $P_u = (G_u^{\Re}, Q_u^{\Re}, R_u^{\Re}, \omega, \rho, \pi)$;
12 **return** P_u;

locations detection. Then, the extraction of movements (line 3) corresponds to aggregate the trajectories with respect to their start-end locations. The filtering of regular locations (line 5) is realized through an effective technique named *knee* method [27], which sets a minimum frequency threshold in a data-driven way (basically it finds the knee of the curve drawn by sorting the locations' frequencies) and discards the locations below it. The steps followed by $filterOccasional$ (line 6) is described in the previous section. The extraction of the location-points and routines (lines 7-8) corresponds to computing the medoid of a set of objects [27]. In particular, the distance function used in $getLocationsPoints$ ($pointDist$) is implemented as the well-known *spherical distance*, while for $getRoutines$ ($trajDist$) we adopted the *IRD* function between trajectories [11]. Finally, the behavioral functions (line 10) are computed by counting the users' presences, according to Def. 8, at intervals of 60 seconds, which appears to be a negligible discretization for most applications.

C. Reproducing Personal Mobility Agendas

In this section we present the procedure we propose for reproducing the mobility agenda exploiting model P_u, which is named *RAMA – Routinary Actions Mobility Agenda*.

Alg. 2 illustrates the logic behind our approach. An empty agenda $\mathbb{A}$ is initialized and, as initial state, we assume the user is not moving (line 1). Then, the initial location is *chosen* among those in $\mathcal{L}^{\Re}$, based on the frequency of the locations ω and the probability of presence ρ at time 0 (line 2). Different ways of choosing locations and movements are discussed later in this section, together with the role of parameter d. For the sake of simplicity, yet without loss of generality, we let the agenda start from midnight, i. e., ts=0. However, this setting can be personalized either according to the behavior of the user or following the desiderata of the analyst.

The loop (lines 3–19) reproduces $\mathbb{A}$ by iteratively adding the predicted/simulated positions for each regular interval τ (line 19). The number of iterations depends on the time tick τ. Υ (line 3) indicates the maximum time (in the time unit adopted), where Υ mod τ=0. Since the *second* is a sufficiently small unity, we use it as default, and we have Υ=86400.

If the user is not moving we check if, according to P_u at time ts, the user is expected to stay in the current location L_{cur} (line 5). If that is the case, function $isInLoc$ returns

Algorithm 2: $reproduceAgenda(P_u, d, \tau)$

1 $\mathbb{A} \leftarrow \emptyset$; $not\_moving \leftarrow True$;
2 $L_{cur} \leftarrow initLoc(\omega, \rho, Q_u^{\Re}, d)$;
3 **for** ts **in** $0, \tau, 2\tau, 3\tau, \ldots, \Upsilon$ **do**
4 **if** not_moving **then**
5 $not\_moving \leftarrow isInLoc(\rho, L_{cur}, ts, d)$;
6 **if** not_moving **then**
7 $lon, lat \leftarrow getPosition(Q_u^{\Re}, L_{cur}, d)$;
8 **else**
9 $L_{nxt} \leftarrow nextLoc(\pi, L_{cur}, ts, d)$;
10 $M_{cur} \leftarrow getMov(\mathcal{M}_{\sqcap}{}^{\Re}, L_{cur}, L_{nxt}, d)$;
11 $lon, lat \leftarrow getPosition(R_u^{\Re}, M_{cur}, ts, d)$;
12 **else**
13 $not\_moving \leftarrow arrivedLoc(M_{cur}, L_{nxt}, ts, d)$;
14 **if** not_moving **then**
15 $L_{cur} \leftarrow L_{nxt}$;
16 $lon, lat \leftarrow getPosition(Q_u^{\Re}, L_{cur}, d)$;
17 **else**
18 $lon, lat \leftarrow getPosition(R_u^{\Re}, M_{cur}, ts, d)$;
19 $\mathbb{A} \leftarrow \mathbb{A} \cup (lon, lat, ts)$;
20 **return** $\mathbb{A}$;

true and the position of the location point of L_{cur} is extracted (line 7). Otherwise, the user will leave L_{cur} to reach L_{nxt}, which is chosen by function $nextLoc$ according to π (line 9). Then, the movement M_{cur} to take is selected (line 10) and the position w.r.t. its routine is computed (line 11).

If, instead, the user is moving (line 12), then the algorithm checks if it is arrived to the destination location L_{nxt} (line 13). If this is the case (line 14), the current location is updated (line 15) and the novel location point is extracted (line 16). Otherwise, the algorithm updates the position based on the routine followed by the current movement (line 18).

Taking Different Choices. For each function in Alg. 2 using ω, ρ, or π there are two possible *choices*. A *deterministic* one, for which the function selects the result with the maximum probability value; and a *probabilistic* one, for which the function selects the result randomly, according to the specific probability distributions. For example, if the probability of being in a location at a certain time is 0.73, then in the first case the result is that the user should be in the location, while in the second case the choice is taken randomly using 0.73 and 0.27 as weights. We refer to the two versions with the names *dRAMA* (deterministic) and *pRAMA* (probabilistic).

Considering Different Days. The approach described so far models a unique, 24-hour typical day of a user and can be employed for simulating/predicting the agenda of any date. However, by assuming an independence of the behavior between consecutive days, the approach allows to calculate the personal mobility model on specific subsets of H_u defined by some notion of similarity. For example we could construct 12 separate monthly models $P_u^1, \ldots, P_u^{12}$, one for each different month of the year, 7 day-of-week models, or just weekdays vs. weekends models. Such specialized models are expected to better capture patterns which are typical of the subset analyzed. For instance, the fact that a user is attending a class

City	#Users	#Trajs	#Locs	#Moves	RG(km)	Entropy	Build Time(s)
Rome	1000	219.8	62.1	106.3	7,078.9	0.63	56.74
London	1000	202.1	56.3	90.2	6,995.4	0.61	47.79
Boston	1000	174.1	46.7	85.7	13,870.1	0.66	36.70
Beijing	120	55.6	25.3	22.7	23,371.2	0.65	4.01

TABLE I: Datasets and Models Statistics: mean of the number of trajectories, locations, movements, radius of gyration (RG), locations support entropy (Entropy) and building time (in seconds).

City	Ratio Traj	#Locs	#Moves	RG(km)	Ratio RG	Entropy
Rome	0.51	6.24	18.37	4,638.12	0.65	0.72
London	0.53	5.62	15.87	4,623.21	0.66	0.69
Boston	0.51	5.82	15.03	8,634.54	0.62	0.72
Beijing	0.48	3.79	3.38	4,165.80	0.17	0.71

TABLE II: Datasets and Models Statistics: same statistics of Tab. I calculated on the regular mobility network, ratio of regular trajectories, and ratio of regular radius of gyration over the standard one.

of music on Wednesday afternoon twice in a month could be completely discarded by the default model, while it would get the appropriate relevance in a day-of-week model. Therefore, the functions in Alg. 2 employ d to select the appropriate model for date d, and its corresponding locations, movements and behavioral functions. We remark, as possible downside, that an abuse of specialized models could lead to *overfitting* problems. If not specified, in the rest of the paper we assume to use the default version of *RAMA*, flattened on the 24 hours.

V. Experiments

In this section we evaluate the performance of different versions of *RAMA*. We test them on real distinct mobility GPS datasets, and compare them against several baselines[1].

A. Datasets

We performed our experiments on real GPS traces collected for insurance purposes by *Octo Telematics*. To prove the effectiveness of the proposed approach on different cities, we used the car travels performed in geographical areas containing the cities of *Rome*, *London* and *Boston*. For each city we considered the traces of 1,000 users active in Jan 2015 - Dec 2015.

Moreover, in order to make our experiments reproducible, we analyzed the performance also on the Geolife dataset [28] containing the movements in *Beijing* of 120 users (all those with more than 20 trajectories) in a period of over 4 years[2].

These datasets cover a broad range of users' movements, including not only home-work routines but also movements leading to shopping centers, to sport activities, restaurants, etc.

B. Models Analytics

In this section we report a brief analysis of the models extracted using *RAMA* on the different datasets. Tab. I illustrates the mean of the sizes of the components of P_u with respect to the complete mobility network G_u. The users in *Rome* have the highest number of trajectories, locations and movements among the *Octo* datasets. On the other hand, due to the different data sources, the users in *Beijing* are defined by less information. Moreover, we report in Tab. I two classical

[1]The Python code is available at https://github.com/riccotti/MobilityAgenda

[2]Geolife dataset https://goo.gl/EPCVw9

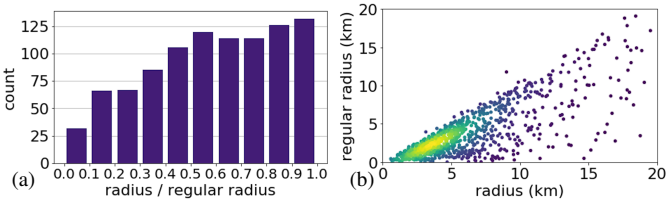

Fig. 2: Models Analytics: *Rome* dataset, *(a)* histogram of the radius of gyrations ratio, *(b)* scatter plot of radius of gyration and regular radius of gyration (the lighter the points the higher the density).

mobility indicators: the *radius of gyration* (RG) [17] and the normalized *entropy* [29] on the locations support ω. They explain the user's predictability with respect to the distance traveled and presences in locations. The higher are these numbers the less predictable is the user. The users in *Boston* and *Beijing* appear less predictable than those in *Rome* and *London*. The last column of Tab. I shows the average *building time* that is less than a second for a typical user. Since the personal data of each user can be treated independently from the others, the models building process is easily parallelizable.

In Tab. II we report the same statistics of Tab. I with respect to the regular mobility network $G_u^{\Re}$. On average half of the trajectories are discarded by *filterOccasional* (Alg. 1): half of the user's mobility passes through regular locations and movements. The regular locations are about one tenth of all the locations, and the regular movements about one fifth of all the movements. The latter values are consistently lower than the ratio of regular trajectories, suggesting that typically a user visits a few different locations following not very diversified movements. Moreover, it is interesting to observe the decreasing of the regular RG with respect to RG in Tab. I: the reduction is similar (around 60% of the original) in the first three cities and it drops in Beijing. On the other hand, the Entropy increases. This happens because if only the regular locations are considered, then the presences among them are more balanced. Hence, typically, the regular radius is lower than the complete one, while the regular entropy is higher. This confirms that we typically live in small well defined areas where the presences are balanced among few regular locations.

Finally, following [17], we analyze the relationship between radius of gyration and regular radius of gyration to verify if the users can be classified as *explorers* or *returners*. The results reported in Fig. 2 contradict this thesis: *(a)* the distribution of the radius ratio is not bi-modal but it follows a skewed right normal distribution, and *(b)* the scatter plot verifies that there are not splits along the x-axis and the bisector of the first quadrant. Thus, it emerges that users have a short range of degrees of "returnerness". This happens because the number of regular locations is not fixed at the same value for all users as in [17], but it is personally tuned by *getRegularLocations* (Alg. 1). This means that from her individual perspective, every user is a returner, while when the number of regular locations is fixed and a collective perspective is imposed by the analyst, then the explorer-returner dichotomy emerges.

C. Evaluation Measures

In this section we present the measures used to evaluate the agenda reproduction, i.e., implementing function $agendaDist$.

We underline how the agenda reproduction problem is very challenging, due to various issues: *(i)* users do not move every time exactly in the same period of the day (at least not exactly); *(ii)* movement speed is not constant during the travel, even following the same trajectory; *(iii)* possible errors deriving from spatial sampling of the data could influence the predicted position. Hence, it is reasonable to consider a set of tolerances.

We use spt_{tol} and tmp_{tol} to describe the spatial and temporal tolerances which generate a spatio-temporal area around the real point. This area contains all the values considered correct for the reproduction problem. Given the reproduced position $p \in \mathbb{A}$ at time $p.ts$, its error $err(p, A)$ with respect to the real agenda A is computed as $min\{pointDist(p, p')|p' \in A \wedge |p'.ts - p.ts| \leq tmp_{tol}\}$. p is correct if $err(p, A) \leq spt_{tol}$.

Let A be the real agenda that we want to replicate and $\mathbb{A}$ the reproduced agenda, given the spatio-temporal tolerances spt_{tol} and tmp_{tol}, we define the following evaluation measures:

- *accuracy* is the relative number of correct points in $\mathbb{A}$, i.e. $|\{p \in \mathbb{A}|err(p, A) \leq spt_{tol}\}|/|\mathbb{A}|$.
- *error* is the average error over all points of the reproduced agenda, i.e. $avg_{p \in \mathbb{A}} err(p, A)$.

D. Baselines

In the literature there are various methods trying to model personal mobility (see Sec. II), but only very few of them take into account the problem of reproducing the entire mobility as a continuous flow. For this reason we generalize the approaches in the literature as *baselines* and we compare them in the agenda reproduction against *RAMA*. The strategy most frequently adopted in literature is to model the user mobility using spatio-temporal cells with dimensions $grid_{spt} \times grid_{spt} \times grid_{tmp}$. The spatio-temporal mobility of the user is discretized into cells, counting the number of GPS points each of them, i.e., the *support* of the cell. The cells' supports are used as probabilities to evaluate the presence of a user at a certain time in a certain area.

GRID. Starting from the most supported spatio-temporal cell (generally containing "home"), *GRID* maintains the user in the current cell until the probability of moving to an adjacent cell is higher. Then the user moves from cell to cell until it reaches a stop cell, and so on. Due to the grid, the coordinates returned for the agenda *GRID* are the centers of the cells.

We name *dGRID* the *deterministic* version selecting the cell with the highest probability, and *pGRID* the *probabilistic* version selecting the cell according to the probability distribution.

The main differences between *GRID* and *RAMA* are that *(i) GRID* markedly discretizes time and space losing the continuity of the behavior and imposing unnatural thresholds, *(ii)* the precision and reliability of *GRID* is affected by the parameters $grid_{spt}$, $grid_{tmp}$. Thus, cells which are too small could split a user location while cells which are too large could not provide a sufficient precision for the agenda, for example by collapsing a movement into a single cell. Moreover, the same parameter setting could be not correct for different users [25].

Symbol	Description	Values
τ	time clock	$\{60, \mathbf{300}, 600, 900, 1800, 3600\}$ sec
spt_{tol}	spatial tolerance	$\{50, 100, \mathbf{250}, 500, 1000\}$ meters
tmp_{tol}	temporal tolerance	$\{60, \mathbf{300}, 600, 900, 1800, 3600\}$ sec
$grid_{spt}$	grid spatial length	$\{50, 100, \mathbf{250}, 500, 1000\}$ meters
$grid_{tmp}$	grid temporal length	$\{900, \mathbf{1800}, 3600\}$ sec

TABLE III: The complete set of parameters used in the experiments. The set of values in bold represents the default setting.

RAND. The *random* method produces a random agenda. It starts from the most supported cell. Then the probability of remaining in the current cell decreases while the time flows. At a certain time *RAND* moves randomly to one of the eight adjacent cell simulating a movement. The movement continues for a length extracted from a normal distribution with mean and standard deviation equal to those of the trajectories of the user. Then the user reaches a staying cell and the procedure is repeated for a number of times, extracted from a normal distribution with mean and standard deviation equal to those of the number of trajectories daily performed by the user.

HOME. The *HOME* method returns as agenda always the most frequent location, that is very likely the home location.

HOWO. The *home-work* method uses the two most frequent cells typically containing home and work. It returns as agenda the center of the home-cell until the probability of being in the work-cell is higher, and viceversa.

It is important to notice that *HOME* and *HOWO* never generate movements in the agenda, but always staying positions.

E. Experimental Setting

For each user u in each city, we build the model P_u on the history $H_u^{d',d''}$, and we reproduce the agendas $\mathbb{A}_u^d(\tau)$ for $d \in \{d''+(i \text{ days})|i \in [1, 7]\}$. In particular, for *Rome*, *London* and *Boston* we build the *RAMA* models and the models of the baselines on the first four month, while for *Beijing* on the first year as the traces are not so accurate as for the other cities, and we test the performance on the subsequent week. Obviously, the deterministic models *dRAMA*, *dGRID*, *HOME*, *HOWO* always produce the same agenda for each date d. The evaluations reported in the rest of the paper are an aggregation over all the reproduced agendas of all the users. With respect to the clock tick τ, spatio-temporal tolerances spt_{tol}, tmp_{tol} and grid dimensions $grid_{spt}$, $grid_{tmp}$ we tested the values reported in Tab. III. When not specified, we report the results obtained by using the values in bold in Tab. III.

F. Comparing with Baselines

In this section we compare *RAMA* with the baselines for all the datasets. Fig. 3 reports the comparison for *Rome*. Either varying $grid_{spt}$ (and correspondingly spt_{tol}) or $grid_{tmp}$ (and correspondingly tmp_{tol}) *pRAMA* has the highest accuracy (Fig. 3 *(a, b)*). This highlights that for *RAMA* the probabilistic approach is better than the deterministic one. Fig. 3 *(c)* depicts the comparison of the performance when varying the time tick τ. As expected, the lower is the value of τ, the worse is the performance, because for high values of τ few comparisons

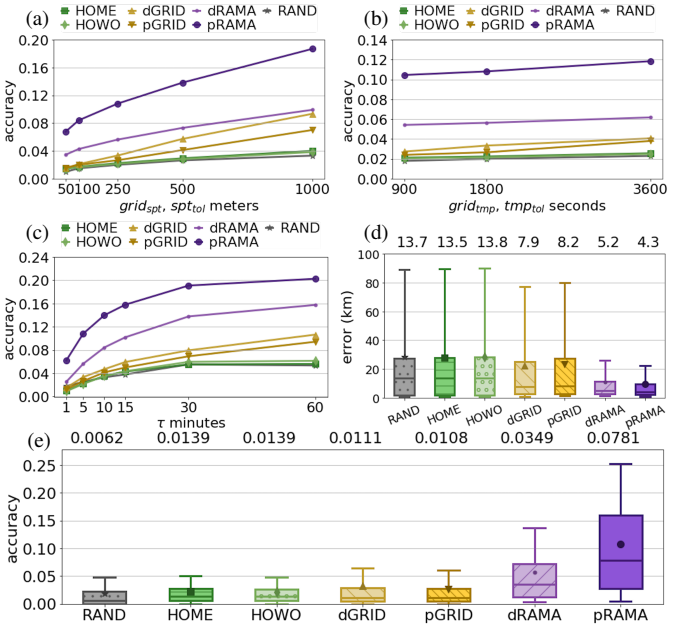

Fig. 3: Comparing with Baselines: *Rome* dataset. The values on the top of *(d, e)* corresponds to the medians of the boxplots.

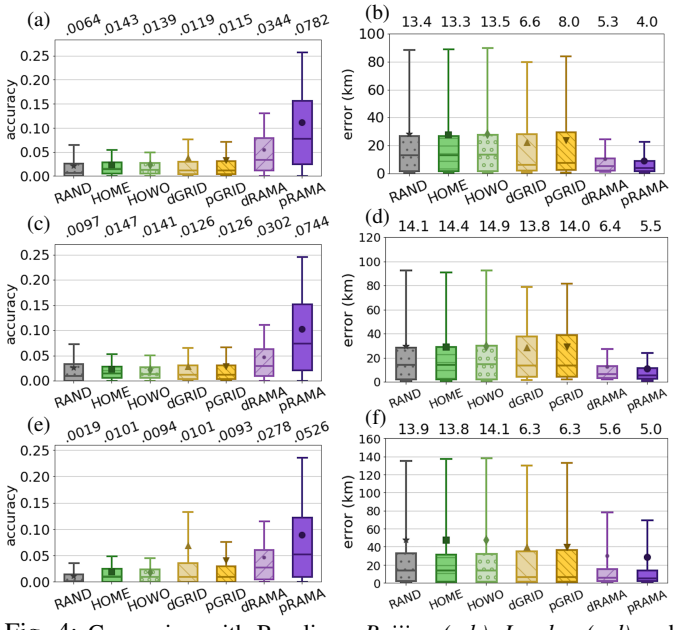

Fig. 4: Comparing with Baselines: *Beijing (a,b)*, *London (c,d)* and *Boston (e,f)* datasets with default parameter setting. The values on the top of the figures corresponds to the medians of the boxplots.

between the real and the reproduced agenda are performed and consequently, together with the fact that we spend the largest part of our time staying in a location rather than moving between two locations, it is easier to obtain an higher accuracy. Besides these considerations, *pRAMA* reproduces the best agendas. Fig. 3 *(d, e)* illustrates more in detail through a set of boxplots the distribution of the error and of the accuracy with respect to the default parameter setting (see Tab. III).

In Fig. 4 we report the same boxplots of Fig. 3 *(d, e)* for *Beijing*, *London* and *Boston* dataset. The performance are very similar among all the datasets with the exception of *Boston* where all the approaches have a slightly lower accuracy. This confirms that, as observed in Sec. V-B, the predictability of the users of *Boston* is lower than those of the users of the other cities. However, *pRAMA* is resulting to be the best approach not only with respect to the baselines but also across different datasets which are the proxies of the mobility of very different cities characterized by different types of drivers. Also *dRAMA* has remarkable high performance but the small weighted perturbation used to differentiate the agendas for the reproduced dates allows *pRAMA* to be the best performer.

We remark that the overall "low" performance is mainly due to the complexity and difficulty of the problem faced, rather than to any degree of negligence of the approaches.

In the following we report the evaluations only for *Rome* and *Beijing*, as they are very similar for the different cities.

G. Considering Occasional Locations

In this section we show the effectiveness of removing occasional mobility from G_u, and considering in P_u only regular locations $L_u^{\Re}$ and movements $M_u^{\Re}$. We name *probabilistic Complete Actions Mobility Agenda* (*pCAMA*) the method using G_u instead of $G_u^{\Re}$. Fig. 5 shows how the variability introduced by all the locations and movements negatively affects the

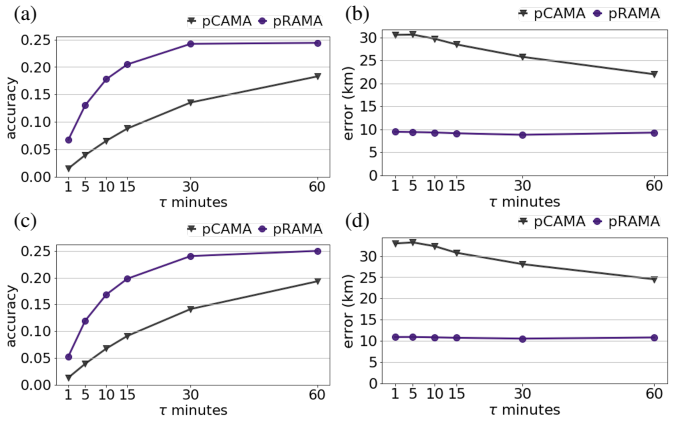

Fig. 5: Considering Occasional Locations: *Rome (a,b)* and *Beijing (c,d)* dataset, accuracy and error varying the clock tick τ.

performance both in *Rome (a,b)* and Beijing *(c,d)*: *pRAMA* has the highest accuracy and lowest error.

H. Models Considering Different Days

In this section we evaluate the performance of *pRAMA* using models realized considering different dates (see Sec. IV-C). In particular, we analyze the behavior of the following versions:

- *24-pRAMA* is the approach flattened on the *24 hours* observed so far: all the dates in H_u are used to build P_u.
- *7d-pRAMA* is the *seven weekdays* approach considering separately each day of the week: seven different models P_u^i $i \in [1, 7]$ are realized (Monday, Tuesday, ..., Sunday). Then the model P_u^i is selected according to the date d reproduced: e.g. if d is a Monday P_u^1 is used.
- *we-pRAMA* is the *weekend* approach considering separately working days (Monday-Friday) and the weekend (Saturday-Sunday): P_u^{we} is used to reproduce the weekends, and P_u^{wd} is used to reproduce the weekdays.

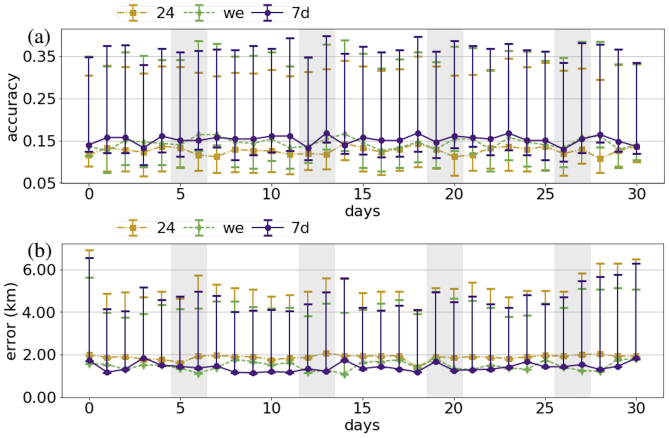

Fig. 6: Models Considering Different Days: *Rome* dataset. Evaluation for each day of the tested month. Gray boxes highlight weekends.

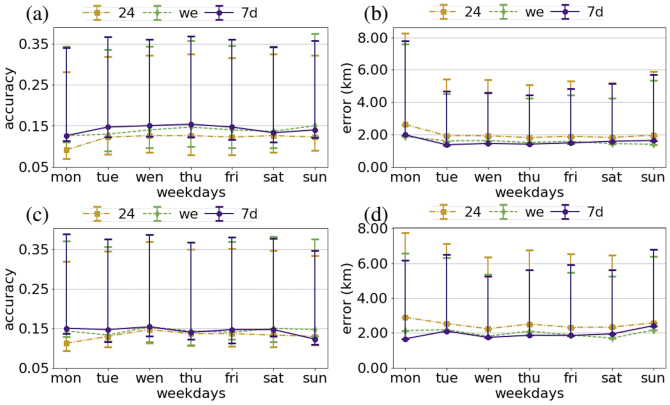

Fig. 7: Models Considering Different Days: *Rome (a,b)* and *Beijing (c,d)* datasets. Evaluation for each weekday of the tested month.

We report in Fig. 6 the performance of the three versions of *pRAMA* on *Rome* with the evaluation of the agendas reproduced for the subsequent month, i.e., we compare A_u^d and $\mathbb{A}_u^d(\tau)$ for $d{\in}\{d''{+}(i \text{ days})|i \in [1,31]\}$. We underline that given the training period $H_u^{d',d''}$ each of the seven models P_u^i of *7d-pRAMA* is extracted using one seventh of the dates used for the model P_u of *24-pRAMA*, i.e., the training dates of *7d-pRAMA* are remarkably less than those of *24-pRAMA*.

The performance of the three versions are mostly stable along the period analyzed. *7d-pRAMA* shows a reproduction refinement with little improvements with respect to *24-pRAMA*. *24-pRAMA* suffers from the replication especially in the weekends (gray boxes). By analyzing the error we observe that *we-pRAMA* often has the lowest values in the weekends. This indicates that considering Saturdays and Sundays together helps more in reproducing the weekends than considering them separately.

This result is stressed in Fig. 7 which illustrates the same experiment for *Rome* and *Beijing* grouping the evaluation with respect to days of the week. Regarding the accuracy, we observe how the purple full line of *7d-pRAMA* remains constantly below the green dashed line of *we-pRAMA* with the exception of the Sunday. Another aspect highlighted by this plot is that the variability of *7d-pRAMA* is higher both in accuracy and error than that of *we-pRAMA*. This is due to the fact that, as

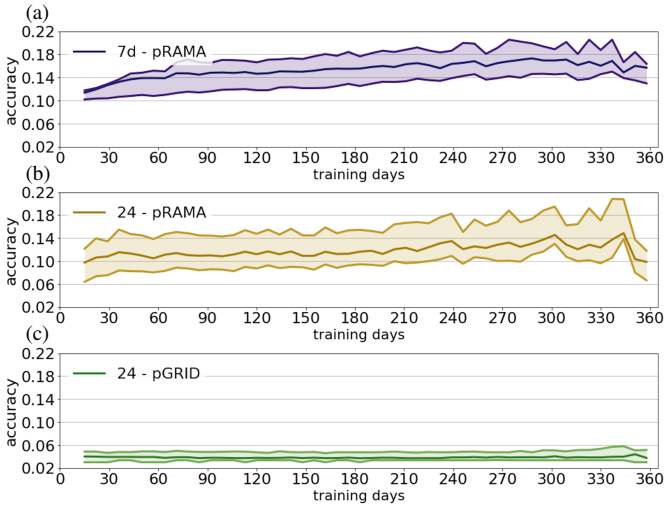

Fig. 8: Learning Curve: *Rome* dataset, accuracy varying the training history $H_u^{d',d''}$ where $d''{\in}\{d|d{=}d'{+}(i \text{ days}), i \in [1,360]\}$.

already mentioned, the seven models of *7d-pRAMA* are built on less days than the two models of *we-pRAMA*.

I. Learning Curve Evaluation

In this section we study the learning curve of *RAMA*, showing how the accuracy of *7d-pRAMA*, *24-pRAMA* and *24-pGRID* on *Rome* varies by enlarging the time period covered by $H_u^{d',d''}$. We notice that the accuracy of *7d-pRAMA* (Fig. 8 *(a)*) slowly but constantly increases. This is due to the already mentioned fact that in *7d-pRAMA* seven distinct models P_u^i are built and, every time that the training set grows there are more Mondays, Tuesdays etc. On the other hand, *24-pRAMA* (Fig. 8 *(b)*) grows more slowly than *7d-pRAMA*. *24-pRAMA* reaches a stability in the learning curve much earlier than *7d-pRAMA*, yet it probably fails to consider some important but less regular movements and locations. Finally, *24-pGRID* remains stable when varying the training period.

J. Semi-Supervised Agenda Reproduction

The agenda reproduction problem, and consequently the models defined, are completely *unsupervised*. Following the experience of [30], in this section we analyze the performance of an agenda reproducer that is occasionally provided with the real position of the user. Hence, we study the *semi-supervised agenda reproduction problem* where at intervals of $\bar{\tau} > \tau$ the real position of the user is provided to the agenda reproducer. More precisely, every $\bar{\tau}$ time units the real position together with the fact that the user is moving or not is provided to *pRAMA* in a sort of "reinforcing reproduction". The semi-supervised version of *pRAMA* reacts by correcting the current positioning with the closest routine or location in P_u.

Fig. 9 shows the reinforcing effect when varying $\bar{\tau}$. As expected, the *semi-supervised pRAMA* markedly overtakes the *unsupervised* one. A very low reinforcing rate is sufficient to distinctly help the agenda reproduction, and possibly to nowcast and reschedule in real time the agenda of a user.

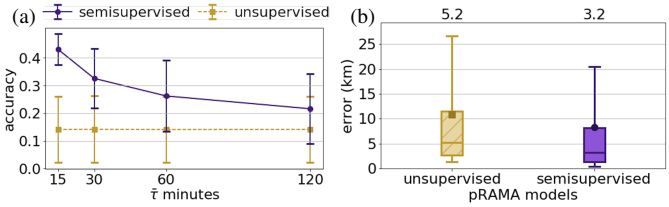

Fig. 9: Semi-Supervised Agenda Reproduction: *Rome* dataset, accuracy and error varying the reinforcing rate $\bar{\tau}$. The values on the top of *(b)* corresponds to the medians of the boxplots.

VI. Conclusion and Future Applications

We have shown how the personal mobility agenda can be exploited to generate future plans for the user's activities. In Sec. V, using the typical evaluation measures from the prediction field, we evaluated the quality of such plans against the real agenda of the users, i.e., the ground truth. The results have shown that the probabilistic model *pRAMA* is slightly better than the deterministic version *dRAMA* and that the weekly *(7d)* model captures better the user behavior. More precisely, the best solution seems to separate the models for each day of the week and collapse the weekend in a single one. Moreover, the approach has shown to be very flexible and able to adapt to various city contexts obtaining comparable results over *Rome*, *London*, *Boston* and *Beijing*. The agenda reproduction obtains a good performance. However, they are significantly improved if real time updates about the real position of the users are provided. This shows that the model is able to reconsider the decisions when new information comes.

In this work we have treated only car trajectories, however, the model presented can be employed also for multi-modal movements (cars, bike, foot, etc.) by obtaining traces and modes from services like Google Location History and building considering diversified probabilities for the behavioral functions for each transportation mode. Moreover, we believe that reproducing the personal mobility agenda is only an example of the applications enabled by the *personal mobility model* defined in Sec. IV-A. It represents a way to compress the mobility maintaining intact the real semantic of the locations and movements. We believe (and it will be part of our future works) that pushing the proposed personal mobility model in a simulator engine such as *SUMO* [31] or *MATsim* [32] will enable these systems to re-create accurate reproductions of the reality not only at individual level, but most importantly at the collective level. This will provide more realistic baselines for application scenarios or what-if analyses. Another application we will explore is the generation of *fingerprints* of a city by using the individual models and their common characteristics, to compare different areas and understand how the similarity is representative of common infrastructures or other socio-economic factors. In addition we are studying how to consider ethical aspects during the model construction and/or in the final applications using a privacy-by-design approach.

Acknowledgment

This work is partially supported by the European Project 654024, *SoBigData*, http://www.sobigdata.eu.

References

[1] J. Schiller and A. Voisard, *Location-based services*. Elsevier, 2004.
[2] Y. Zheng, L. Zhang, Z. Ma, X. Xie, and W.-Y. Ma, "Recommending friends and locations based on individual location history," *TWEB*, 2011.
[3] J. Bao, Y. Zheng, and M. F. Mokbel, "Location-based and preference-aware recommendation using sparse geo-social networking data," in *SIGSPATIAL*. ACM, 2012, pp. 199–208.
[4] D. Quercia, R. Schifanella, and L. M. Aiello, "The shortest path to happiness: Recommending beautiful, quiet, and happy routes in the city," in *HT*. ACM, 2014, pp. 116–125.
[5] R. Guidotti and P. Cintia, "Towards a boosted route planner using individual mobility models," in *International Conference on Software Engineering and Formal Methods*. Springer, 2015, pp. 108–123.
[6] M. Berlingerio, B. Ghaddar, R. Guidotti, A. Pascale, and A. Sassi, "The graal of carpooling: Green and social optimization from crowd-sourced data," *TRC*, vol. 80, pp. 20–36, 2017.
[7] R. Guidotti, M. Nanni, S. Rinzivillo, D. Pedreschi, and F. Giannotti, "Never drive alone: Boosting carpooling with network analysis," *Information Systems*, vol. 64, pp. 237–257, 2017.
[8] M. C. Gonzalez, C. A. Hidalgo, and A.-L. Barabasi, "Understanding individual human mobility patterns," *Nature*, vol. 453, 2008.
[9] C. Song, Z. Qu, N. Blumm, and A.-L. Barabási, "Limits of predictability in human mobility," *Science*, vol. 327, no. 5968, pp. 1018–1021, 2010.
[10] F. Calabrese *et al.*, "Human mobility prediction based on individual and collective geographical preferences," in *ITSC, IEEE*, 2010, pp. 312–317.
[11] R. Trasarti, R. Guidotti, A. Monreale, and F. Giannotti, "Myway: Location prediction via mobility profiling," *Information Systems*, 2015.
[12] R. Trasarti, F. Pinelli, M. Nanni, and F. Giannotti, "Mining mobility user profiles for car pooling," in *SIGKDD*. ACM, 2011, pp. 1190–1198.
[13] S. Rinzivillo, L. Gabrielli, M. Nanni, L. Pappalardo, D. Pedreschi, and F. Giannotti, "The purpose of motion: Learning activities from individual mobility networks," in *DSAA*. IEEE, 2014, pp. 312–318.
[14] R. Guidotti, R. Trasarti, and M. Nanni, "Tosca: two-steps clustering algorithm for personal locations detection," in *SIGSPATIAL 2015*, p. 38.
[15] K. K. Jahromi, M. Zignani, S. Gaito, and G. P. Rossi, "Simulating human mobility patterns in urban areas," *Simulation Modelling Practice and Theory*, vol. 62, pp. 137–156, 2016.
[16] L. Pappalardo and F. Simini, "Modelling individual routines and spatio-temporal trajectories in human mobility," *arXiv:1607.05952*, 2016.
[17] L. Pappalardo, F. Simini, S. Rinzivillo, D. Pedreschi, F. Giannotti, and A.-L. Barabási, "Returners and explorers dichotomy in human mobility," *Nature communications*, vol. 6, 2015.
[18] A. T. Ihler and P. Smyth, "Learning time-intensity profiles of human activity using non-parametric bayesian models," *NIPS*, vol. 19, 2007.
[19] S. Lee *et al.*, "Next place prediction based on spatiotemporal pattern mining of mobile device logs," *Sensors*, vol. 16, no. 2, p. 145, 2016.
[20] H. Wang *et al.*, "Location recommendation in location-based social networks using user check-in data," in *SIGSPATIAL*, 2013, pp. 374–383.
[21] E. Cho *et al.*, "Friendship and mobility: user movement in location-based social networks," in *SIGKDD*. ACM, 2011, pp. 1082–1090.
[22] T. Anagnostopoulos *et al.*, "Mobility prediction based on machine learning," in *MDM*, vol. 2. IEEE, 2011, pp. 27–30.
[23] V. W. Zheng *et al.*, "Collaborative location and activity recommendations with gps history data," in *WWW*. ACM, 2010, pp. 1029–1038.
[24] R. Guidotti, "Personal data analytics - capturing human behavior to improve self-awareness and personal services through individual and collective knowledge," Ph.D. dissertation, University of Pisa, 2017.
[25] E. Keogh, S. Lonardi, and C. A. Ratanamahatana, "Towards parameter-free data mining," in *SIGKDD*. ACM, 2004, pp. 206–215.
[26] J. R. Norris, *Markov chains*. Cambridge university press, 1998, no. 2.
[27] P. Tan *et al.*, *Introduction to data mining*. Pearson Education, 2016.
[28] Y. Zheng, L. Zhang *et al.*, "Mining interesting locations and travel sequences from gps trajectories," in *WWW, ACM, 2009, 791–800*.
[29] C. E. Shannon, "A mathematical theory of communication," *SIGMOBILE*, vol. 5, no. 1, pp. 3–55, 2001.
[30] B. N. Miller, I. Albert, S. K. Lam, J. A. Konstan, J. Riedl, and J. A. K. J. Riedl, "Movielens unplugged: Experiences with a recommender system on four mobile devices," in *IUI*. Citeseer, 2003.
[31] D. Krajzewicz, "Traffic simulation with sumo–simulation of urban mobility," in *FTS*. Springer, 2010, pp. 269–293.
[32] U. Beuck, K. Nagel, M. Rieser, D. Strippgen, and M. Balmer, "Preliminary results of a multiagent traffic simulation for berlin," *Advances in Complex Systems*, vol. 10, no. supp02, pp. 289–307, 2007.