

Cell phone big data to compute mobility scenarios for future smart cities

Davide Tosi¹

Received: 19 May 2016 / Accepted: 20 June 2017 / Published online: 28 June 2017
© Springer International Publishing AG 2017

Abstract Efficient mobility is a key aspect for the future smart cities. The real added value for smart cities is the real-time optimization of vehicular and public transportation flows to reduce traffic congestions, costs, and emissions. Observing constantly the behaviour of people moving around the city can help policy makers to act promptly and to fix congested flows dynamically. In this paper, we describe from a technical point-of-view an original use of big data (coming from the cellular network of the Vodafone Italy Telco operator) to compute mobility patterns for smart cities. The paper also discusses five innovative mobility patterns that describe different mobility scenarios of the city, starting from how people move around point-of-interests of the city in real time. The mobility patterns have been experimentally validated in a real industrial setting and for the Milan metropolitan city. The study conducted confirmed the quality of the patterns and their importance in smart cities, by showing how cell phone big data can complete other sources of people information. These mobility patterns can be exploited by policy makers to improve the mobility in a city, or by Navigation Systems and Journey Planners to provide final users with accurate travel plans.

Keywords Cellular network data · Big data · Mobility patterns · Smart cities

This paper is an extended version of the BDS2016 accepted best paper: “Big Data from Cellular Networks: Real Mobility Scenarios for future Smart Cities” [1].

✉ Davide Tosi
davide.tosi@uninsubria.it

¹ Università degli Studi dell’Insubria, DISTA - Via Mazzini 5,
Varese, Italy

1 Introduction

Efficient mobility is a key aspect for the future smart cities. The real added value for smart cities is the real-time optimization of vehicular and public transportation flows to reduce traffic congestions, costs, and emissions. Big data coming from heterogeneous sources are the basis to build an ecosystem of services that exposes aggregated and elaborated data with a lot of semantics.

In the literature, the active and passive approaches are the two main ways to collect and expose data: active approaches require ad hoc infrastructure to collect data, while passive approaches exploit existing solutions and infrastructure to collect data passively. For instance, traditional approaches to compute real-time vehicular traffic situations are based on dedicated infrastructures, such as road sensors, inductive loops, closed-circuit televisions (CCTVs), and emergency calls to actively collect data that are elaborated to provide final users with traffic estimations. However, these solutions are expensive and invasive for the habitat. Alternatively, more modern approaches try to exploit passively alternative sources of data, such as social network discussions, GPS data, and cellular network data (i.e. data exchanged between antennas and mobile devices) to compute traffic estimations [22,23]. While approaches based on social networks and GPS data are quite limited in practice (for example, due to the reliability of social discussions or the limited availability of data points collected from GPS sources), cellular network data are pervasive and always available [12–14,19].

In this paper, we focus on the use of an innovative passive approach where 3G signalling is silently collected from the Vodafone Italy (VI) cellular network. In this scenario, mobile users become an alternative source of data to predicting vehicular traffic situations and also to estimating mobility patterns of human mobility. This work is part of a bigger

European project called SUPERHUB (<https://ec.europa.eu>), where also GPS and social networks data are used to derive mobility scenarios.

Mobility patterns describe the behaviour of individual citizens in living the city and in affecting the behaviour of other people [1–4, 10, 15, 17]. In this paper, we discuss new algorithms to collect and elaborate in real time the big data coming from the VI cellular network to compute mobility patterns. These patterns describe different mobility scenarios of the city, starting from how people move around point-of-interests of the city, how people affect traffic and speed profiles of each road of the city, how people commute from the outskirts to the city centre and vice-versa, how people use the subway lines of the city, and how much subway stations are crowded.

The output of the mobility patterns will then be used, for example, by Navigation Systems and Journey Planners to compute the best routes based on real-time traffic and speed profile information, and also by Policy Makers and Policy Simulators for urban planning and to enforce some policies based on traffic data and mobility patterns discovered at runtime, such as to adapt at runtime the frequency of public transports, the timing of traffic lights, or the dimension of congestion-charge areas (C-Areas). The different algorithms and mobility patterns implemented have been experimented in the real industrial setting of VI (i.e. in the production network infrastructure of VI), by using the hundreds of thousands of real mobile users of VI, by elaborating hundreds of Giga Bytes of data at real time (anonymized for privacy and security reasons) and for the big city of Milan.

The paper is structured as follows: Sect. 2 introduces the related work in the field and discusses critically the main differences with our approach. Section 3 discusses the technicalities to compute the statistical models that are at the basis for the mobility patterns described in Sects. 4 (vehicular traffic), 5 (speed profile), 6 (Origin/Destination matrixes), 7 (point-of-interest people flows), and 8 (subway people flows). Section 9 reports on Threats to Validity of this work. We conclude in Sect. 10.

2 Related work

In this section, we introduce the research work available in the area of Floating Car Data, which is at the basis of the scenarios presented in this paper.

Floating Car Data (FCD) is a recent research area that investigates on how to exploit signalling coming from different electronic devices (for example, mobile phones, electronic tool collection systems, and GPS-based receivers) active in private vehicles, in order to derive behavioural mobility patterns (i.e. vehicular speed, directions of travel, travel times, traffic info) in one or more points of the city [11–13, 15, 16].

A first differentiation on the FCD techniques can be done between active and passive monitoring systems. In active monitoring, additional signalling traffic procedures are defined and then gathered from the devices at runtime to derive users' location and position. In passive monitoring, the already available signalling information are silently collected, without impact on the network load, and then properly elaborated to derive users' position.

Another differentiation of the FCD techniques can be done between Floating Cellular Network Data (FCND) and Floating GPS Data. In FCND, every switched-on mobile phone (2G, 3G, 4G) becomes a traffic probe and becomes an anonymous source of information that can be used to determine behavioural mobility patterns by elaborating the position of each mobile equipment (for example, by using triangulation or algorithms that aggregate low-level events coming from the network such as handover, location updates, cell updates, and calls set-up) [19].

Alternatively, in Floating GPS Data, GPS user positioning information is used to derive road traffic data [23–25].

While GPS is more accurate than Cell Net Data, FCND becomes more accurate than GPS in real conditions where GPS data are available less than 10% of the time. Moreover, mobile phones continuously report location events to the cellular infrastructure, thus this type of signalling has no impact on the performance of the mobile phone (for example, battery consumption) as in the case of activating GPS tracking systems.

Regarding Floating Cellular Network studies, the CAPITAL project (Cellular Applied to ITS Tracking and Location) [26] was the first project that exploited mobile phones as traffic probes. Mobile phones were used to estimate roughly traffic conditions limited to eight cellular towers. In our PAPT approach (Passive Approach for Predicting Traffic), we formally and experimentally validate the derived models, and we do not limit the scope of the approach to a small number of cells (for example, as for the Milan area 1900 cells are covered.)

The European STRIP project [21] aims at computing travel-time estimates from GSM signalling messages. The feasibility of the approach has been recently experimented on the French Rhone corridor network, showing good estimations if compared to the data collected by the detectors. In PAPT, we use both 2G and 3G interfaces to improve the accuracy of the estimation and we compute the traffic estimations on the granularity of small areas—instead of roads—(i.e. the city is split into 1200 areas).

More recent papers has been published by Calabrese et al. [17, 18]. In [17], they describe the use of the Enhanced Cell-ID with Timing Advance (TA) algorithm to localize mobile phones and to compute behavioural mobility patterns of the monitored users in Rome. In our approach, we use different cellular network data (probed by the A 2G and Iu-CS

3G interfaces) to predict traffic situations in order to overcome the limits of solutions based on TA algorithms (i.e. the high cost of the probes, and the impact of the localization errors introduced by the TA in urban areas where cells are small). In [18], the authors show a visual representation of Origin/Destination flows to optimize Public Transport. This work is very recent (2016) and in line with the scenario and definitions we described in 2014 in the SUPERHUB European project (<https://ec.europa.eu>).

In [14], an empirical evidence is provided to describe the strong relation between road traffic and 3G signalling, but the paper leaves as future work the implementation of an algorithm that considers jointly: changes in the location updates, increases in the number of call and SMSs, and sudden changes in the number of users in a cell, in order to predict traffic anomalies.

3 The phases of the PAPT approach

Our approach to estimate all the presented scenarios starts from the passive analysis (i.e. network signalling is silently collected from the VI infrastructure) of cellular network data. It is called PAPT: *Passive Approach for Predicting Traffic*. PAPT starts from the assumption that statistical models can describe the correlations between real traffic situations and cellular network events collected by the probes. With this assumption, we can use these models to define all the mobility scenarios presented in the following sections. Hence, it is important to understand the main phases that compose PAPT, starting from the statistical models detection to the use of the models with real-life data.

The PAPT approach is based on four main phases, starting from (1) the detection of correlations and the definition of statistical models for the two sources of data: cellular network data collected by the VI probes, and traffic situations collected by other external sources such as feed rss, Google traffic, or manual sampling, (2) the real-time collection of VI cellular network data, (3) the execution of the statistical models against the real-time data collected by the probes to predict the current traffic situation, and finally (4) the computation and graphical representation of a set of indicators that describe the current traffic situation.

Figure 1 summarizes the phases of the approach, highlighting the phases that occur offline or online.

3.1 Phase 1: deriving statistical models

In this first phase, historical data, which are collected by the probes and stored in a dedicated database (DB), are correlated with other sources of traffic (such as feed rss, Google traffic, social GPS navigation system, Waze (www.waze.com), and manual sampling of the traffic in target points

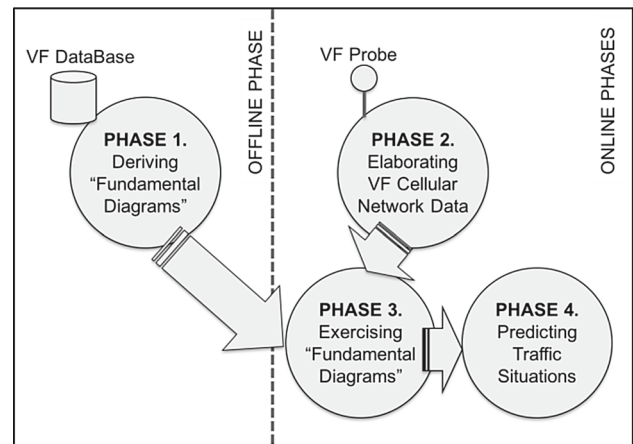


Fig. 1 Phases of the PAPT approach

of the city). These correlations are modelled by means of Regression functions that try to model relationships between dependent variables Y (in our case, the data collected by the other sources of real traffic) and independent variables X (in our case, the data collected by the VI probes). After deriving such models, if an additional value of X is detected (in our case, the data collected at runtime by the probes), it is possible to predict the future value of y (in our case, the actual status of traffic) without having the actual observation of y .

PAPT exploits both Linear regression and also Logistic Regression in case of univariate independent variable X and also in case of multivariate X .

In general, Linear regression produces the slope of a line that best fits a single set of data. For instance, suppose you are interested in projecting the appropriate price for a house in a target area, based on square footage. Using a linear regression formula, you can estimate a price, based on a database of information gathered from existing houses. Logistic regression produces an exponential curve that best fits a set of data that you suspect does not change linearly. Multiple regression is the analysis of more than one set of data (for the independent variable X), which often produces a more realistic projection. In this case, you can perform both linear and exponential multiple regression analyses. Taking the previous example, the appropriate price for a house can be projected focusing not only on square footage, but also on number of bathrooms, age, etc. Using a multiple regression formula, you can describe relationships between the price and all the characteristics of the house. Focusing on our scenario, take as example these data:

In Table 1, Column $\langle Y: \text{traffic_Indexes} \rangle$ lists the indexes of traffic that are observed for a specific area in a specific timeslot by analysing the data coming from the external sources of traffic. The indexes follow the four traffic categories we identified (i.e. 1 = free flow, 2 = high, 3 = congested, 4 = impossible) in compliance with the definitions available in DATEXII [www.datex2.eu].

Table 1 Example of correlating data

Y: traffic_Indexes	X ₁ : #ofLocation_Update_Reqs	X ₂ : #ofHandover
1	10	4
1	12	3
2	25	7
4	160	23
4	210	21
–	–	–

Column <X1: #ofLocation_update_Reqs> lists the number of Location Update Request events that are registered by the probes for a specific area in a specific timeslot. Column <X2: #ofHandover> lists the number of Handover (i.e. the event when an in progress phone call is redirected from its current cell to a new cell) events that are registered by the probes for a specific area in a specific timeslot. In this example, it is clear that a linear relation between the dependent variable Y and the independent variable X1 exists: when the number of Location Update Requests grows, the traffic index increases accordingly. It is also obvious that a multivariate analysis suggests a linear relation between Y and <X1, X2>. Of course, to have models that are statistically significant, we need a data set of hundreds of observations. In case of linear univariate regression, we are interested in computing the following equation:

$$\text{Formula 1: } f(x) = m * x + b$$

where $f(x)$ is the dependent variable as function of the independent variable x ; m is a coefficient for x , and b is a constant. m is computed as:

$$\text{Formula 2: } m = \frac{\sum(x-\bar{x})(y-\bar{y})}{\sum(x-\bar{x})^2}$$

In case of linear multivariate regression, we are interested in computing the following equation:

$$\text{Formula 3: } f(x) = m_1 * x_1 + m_2 * x_2 + \dots + m_i * x_i + b$$

where $f(x)$ is the dependent variable as function of the independent variables x_i ; m_i are the coefficients for x_i , and b is a constant.

The statistics related to the regression we are interested in are the following:

- Se_b : standard error values for the constant b ;
- Se_y : standard error value for y ;
- Se_m : standard error values for coefficient m ;
- R^2 : coefficient of determination. It describes the correlation degree between y and x . If the data set has a perfect correlation $R^2 = 1$. In the worst case $R^2 = 0$;
- Df: degrees of freedom;
- F and t tests: F and t tests verify whether the relationships between y and x are random relations. For example, with a p value=0.05 and Df=6, the critical level of F is

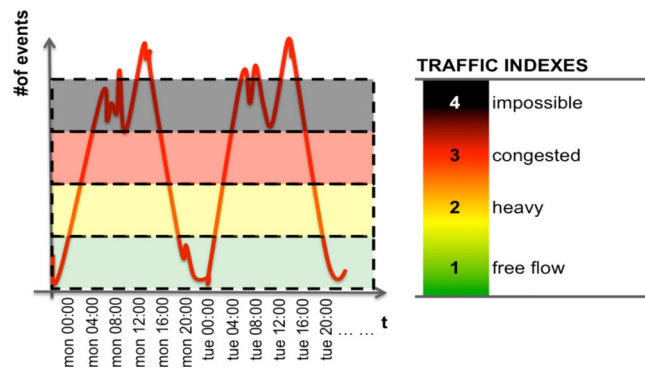


Fig. 2 Time-event diagrams and associated traffic indexes

$F = 4.53$. If the computed value of F is greater than its critical level, the correlation is not a random correlation. The same happens with the t test (with different values);

- SS_{reg} : regression sum of squares;
- SS_{resid} : residual sum of squares.

The output of this phase is firstly a set of significant models (i.e. a set of matrixes) that describe the correlation computed by the regressions for a meaningful sample of cells for the VI network, and secondly a set of time-event diagrams that describe the historical number of network events for each VI cell (i.e. area of the city) and in a specific time-window (for example, in the week), as shown in Fig. 1.

Time-event diagrams (see Fig. 2) are computed by the statistical tool “R” [8] by averaging the historical observations collected by the probes. Correlation data and time-event diagrams are mapped to provide the four levels of traffic (“free flow”, “heavy”, “congested”, and “impossible”) as shown in Fig. 1.

3.2 Phase 2: elaborating VI cell net data

In this second phase, real-time VI cellular network data are gathered by the installed probes (deployed around the VI Network on MSSs to monitor Iu-CS RANAP and GSM-A BSSAP traffic), elaborated (i.e. data are compressed, aggregated and anonymized) by the TAMS Server, and then stored in a dedicated VI server (called Floating Car Data Platform), as depicted in Fig. 4.

The output of the TAMS server is a set of files in .csv format that contains the following network information:

- Timestamp: the elapsed time since 01/01/1970 expressed in seconds;
- Transaction type: the event gathered by the network. The following events are monitored: CM Service Request, Common ID, Paging Response, Location Updating Request, Location Updating Accept, TMSI Reallocation,

Location Report, Relocation Command (temporary), Location Report, Handover;

- LAC: the Location Area Code;
- SAC or CI Identifier: the Service Area Code or the Cell Identifier;
- O-IMSI: the International Mobile Subscriber Identity obfuscated by mean of hashing mechanisms;
- 2G/3G Indicator: UTRAN / GERAN indicator to disambiguate from data coming from the A interface to data coming from the Iu-CS interface.

The records are structured as in the following example:

```
1297868695,3,51504,6360,3E6AF3303D865D23,1
```

The records managed by the TAMS server and stored in the Floating Car Data Platform are elaborated to filter all the cellular events that are not relevant for computing vehicular traffic predictions. To this end, we act in two ways:

1. Detect the mobility modalities associated with each tracked mobile equipment, and filter all the network events that refer to mobile users, which probably are still or are moving on foot or by bike. To do this, we implemented an algorithm that computes at real time the speed of each mobile user between the centroid of the cell with respect to its previous one. Velocity tags are then set and all the tags with current speed < 15 are discarded;
2. Detect and filter ping-pong hops (and network anomalies such as too fast movement between adjacent cells) from the sequence of cells traversed by a target mobile equipment. A ping-pong hop occurs when, given two cells C_a and C_b , a mobile equipment moves from C_a to C_b and back to C_a within an adaptive time-window. For space reason, we do not describe in detail the defined and implemented algorithm.

When all the network events are filtered, the numbers of aggregated events are used to train the statistical models and forecast the current traffic situation.

3.3 Phase 3: exercising statistical models

The correlations detected during Phase 1 are trained with the current number of events to detect the associated level of traffic. Hence, the current number of events is seen as an additional value of X that is used to predict the future value of y (i.e. the actual real-time status of traffic) without having the actual observation of y . For each cell C_i of the VI network, the associated time-event diagram is retrieved from the Floating Car Data Platform and the current number of events is compared with the time-event diagram. For example, if we take as target timeframe Monday 3:00 PM and the number of

computed real-time events is in line with the curve of Fig. 2, the correlated level of traffic is equal to “congested”.

3.4 Phase 4: predicting traffic situations

In the last phase, the maps with traffic situations are generated. To this end, three layers are merged: the target map of the city, the topology of the cells for the VI network, and the layer that describes the traffic index computed at runtime for each cell. Because of traffic indexes can vary at runtime, these maps are updated online, accordingly.

4 Vehicular traffic estimations

4.1 Overview of the scenario

Vehicular Traffic estimations from the VI cellular network provide policy makers with real time and historical vehicular traffic indicators for sub-areas of Milan [4] (see Fig. 3 for the graphical output of our approach.)

As stated in Sect. 3, our new passive approach starts from the assumption that statistical models can describe the correlations between real traffic situations and cellular network events collected by the VI probe. The probe sniffs in real time the events (voice, data, SMS, etc.) generated by the A Interface and Iu-CS Interface of the 2G (GSM) and 3G (UMTS) cellular network. It is important to notice that the adopted probing infrastructure is primarily used for monitoring the VI network quality; hence, this approach does not introduce further costs for the telco operator from an infrastructural point-of-view.

4.2 Technical aspects

In our work, we started by finding significant regression models able to describe these correlations. We sampled six different areas of Milan. The six areas have been selected with the heterogeneity requirement in mind in order to apply the methodology to different areas of the city with different peculiarities (areas characterized by small streets, very large roads, very congested areas, etc.) We used as dependent variable X , the number of network events collected for one hundred different sample data points. We started collecting data points from 12/02/2013 to 29/04/2013 in different hours of the day. The number of network events (#ofEvents) is computed by considering all the transaction types of the Vodafone network, and with time granularity of 2 mins (across the timestamp of the invocation.) As for the independent variable Y , we collected and used the data provided by the InfoBlu source [www.infoblu.it]. InfoBlu is a service that provides real-time traffic data for the most important roads of the Italian network.

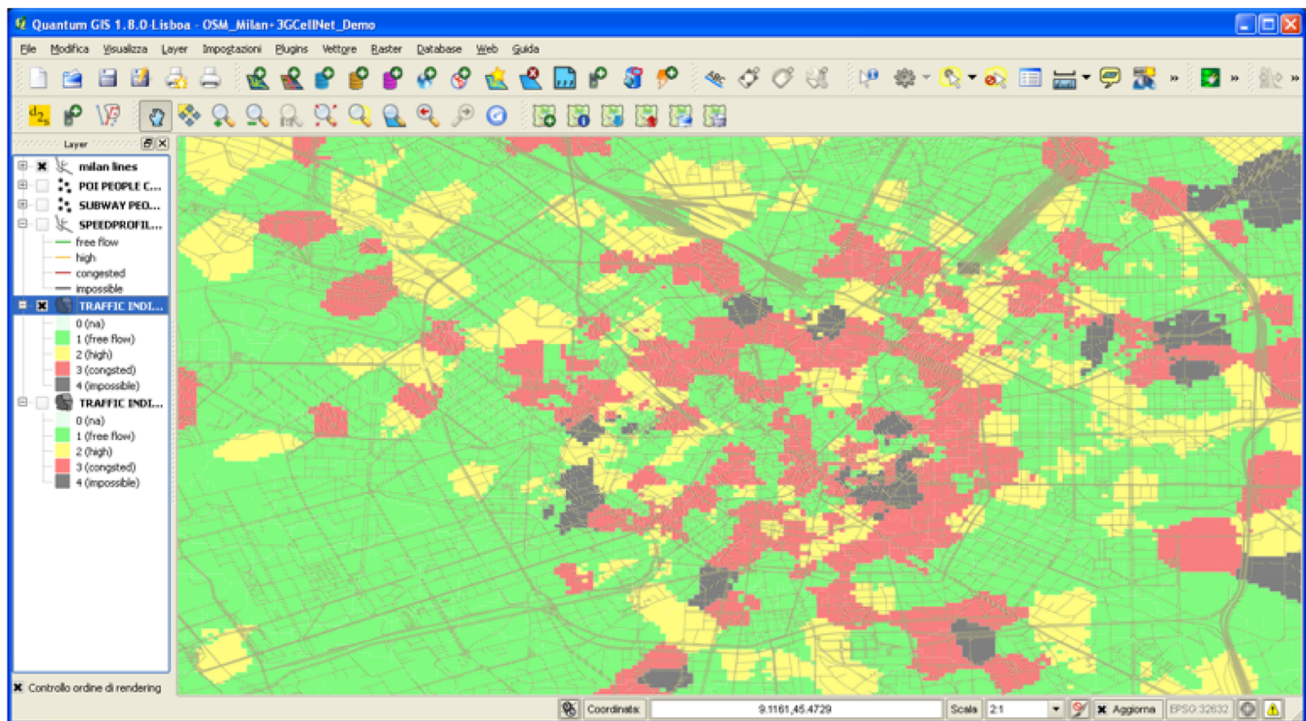


Fig. 3 Output of the vehicular traffic estimator service

We applied linear, loglog, and polynomial regression functions to the six case studies, using the Ordinary Least Squares (OLS) regression [21]. We used the stringent 0.01 as the statistical significance threshold, as is customarily done in empirical studies. Therefore, all the reported models have p value < 0.01 . The normality of the distribution of the residuals, which is a statistical requirement for safely applying OLS regression, was tested by means of the Fisher Test [5]: consistent with our statistical significance threshold, p -values > 0.01 do not allow the rejection of the normality hypothesis.

In this phase, we obtained several models and we selected the most performing one (with a determination coefficient $R^2 = 0.7229$, while all the other models have a determination coefficient $R^2 < 0.5$). If we denote by *TrafficPrediction* the traffic index with a value ranging from 1=fluid to 6=impossible (following the values provided by InfoBlu)¹ and by *#ofNetEvents* the total number of detected VI network events (such as location updates, handovers, timsi relocations), the obtained most statistically significant OLS regression model (polynomial), which describes the correlation between VI cell events and real traffic data for all the areas of the city, is the following (the reader can find additional details on how coefficients are computed in our paper [5].)

¹ It is important to observe that the InfoBlu traffic scale (from 1 to 6) must be normalized to our scale (from 1 to 4).

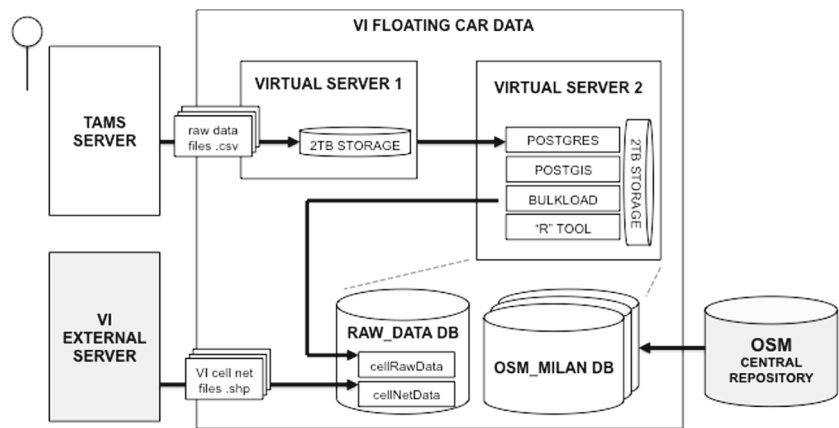
$$\begin{aligned} \text{TrafficPredictionThresholds} = & 2.3652 * \#ofNetEvents^3 \\ & - 18.522 * \#ofNetEvents^2 + 78.896 * \#ofNetEvents \quad (1) \end{aligned}$$

The precision of the fitting is good. The regression sum of squares is $ssreg=41.28$, the residual sum of squares is $ssresid=19.20$ and the value of the standard error is $sev=0.57$. The residuals are normally distributed (i.e. the normality hypothesis cannot be rejected), because the Fisher Test Value F (124.68) is greater than the critical value (7 in case of 100 df_2), thus the null hypothesis H_0 is refused with a probability $p < 0.01$. The determination coefficient is good ($R^2 = 0.7229$) also. So, around 72% of the variability in the degree of traffic prediction is explained by the total number of network events collected by the probe.

This model can then be used to generate in real time the map with vehicular traffic info of Fig. 3. The reader can find additional details in [5].

The current implementation of the VI infrastructure, which elaborates these data to estimate vehicular traffic information, is depicted in Fig. 4. The figure highlights three main sources of data: (1) The data coming from the probe and the TAMS Server (Troubleshooting and Monitoring System) to collect the data coming from the probe and to assemble it in the correct format for transmitting it to the main infrastructure, where data are elaborated to derive the dynamic distribution of the SIM cards in the area monitored by the probes. From the TAMS Server, .csv files with the cellular

Fig. 4 VI infrastructure for the traffic estimation service



network data are pushed to the VI Floating Car Data Server; (2) The data coming from the VI External Server from which the RAW_DATA DB receives the topological network of the VI cells as shape .shp files; (3) The data coming from the OpenStreetMap (OSM) Central Repository from which the OSM_MILAN DB receives the OSM maps [9].

4.3 Experimental activities

We extensively experimented the Vehicular Traffic Estimator in a real environment (i.e. the VI production environment), with thousands of real VI mobile users, and in the crowded city of Milan. The experimentation leads us the possibility to test the infrastructure against millions of network events and GB of data generated in few minutes. With these experiments, we were interested in collecting data to show whether the detected regression model (Formula 1) and the estimations coming from our approach are better than the estimations provided by two of the major traffic info providers in Italy: GoogleTraffic and InfoBlu, when compared against real traffic situations.

To this end, we used the model of Formula 1 to evaluate in real scenarios whether—for a specific area of the city—our estimations better approximate the traffic status than the results provided by the two companies (they build their estimations by using GPS traces.) In order to validate the comparison, we conducted two experiments: the first one for primary roads, and the second one for secondary roads in Milan (for the used classification of roads see [9]). As for the first experiment, we collected 68 real traffic situations by means of the Autostrade.it web site (www.autostrade.it), which provides real-time traffic situations computed by evaluating webcam frames and inductive loops data dislocated in the most relevant area of Milan. In this way, we are able to collect data points that are reliable. As for the second experiment, we collected 40 traffic situations by observing webcam frames for a secondary Milan road and we use the observations as benchmark for our approach.

To evaluate the precision of our approach, we computed the Standard Error (std.err) of each solution, by applying the following equation:

$$\text{STD.ERR.YX} = \sqrt{\frac{1}{(n-2)} \left[\sum (y-y)^2 - \frac{[\sum (x-x)^2 (y-y)^2]}{\sum (x-x)^2} \right]}$$

As for the first experiment, our approach overestimates a little bit the traffic status (the average of traffic indexes from Autostrade.it is equal to 1.59, while the average from our approach is 1.90) and the standard deviation of 0.87. Our estimations are quite good with a std.err=0.58 (a little bit greater than the std.err of GoogleTraffic and InfoBlu of 0.46 and 0.51, respectively). All the three approaches were able to provide estimations for all the 68 observations (unavailability=0%). This experiment clearly shows that for areas where GoogleTraffic and InfoBlu can count on a lot of GPS data (primary roads of the city), our approach is a little bit less precise than the other two sources. However, as for the second experiment (e.g. on a second experiment for a secondary road of the Milan city: Viale Monza Street), our approach is more precise if compared to InfoBlu and GoogleTraffic. In this case, we use as source of real traffic situations, a webcam [7] installed in Viale Monza Street (lat/long: 45.5172/9.2255) to manually evaluate the traffic level. We collected 40 data points spanned in 10 data-times. In this case, the estimations of our approach are very good with a std.err equals to 0.35 (smaller than the std.err of GoogleTraffic of 0.64). InfoBlu was not able to compute traffic predictions for any of the 40 observations, while GoogleTraffic did not provide predictions for 8 out of 40 observations (unavailability=20%), and our approach just for 2 observations due to probe connectivity problems (unavailability=5%). At the time of writing, all connectivity problems are solved with an availability of the service equals to 99.99%. Tables 2 and 3 summarize the results of the experiments conducted in Milan.

Table 2 Vehicular traffic: excerpt of real traffic versus PAPT estimations versus InfoBlu and GoogleTraffic (primary roads in Milan)

Timestamp	Where (Lat/Long)	Real traffic	PAPT	InfoBlu	GoogleTraffic
120/05/2013 at 17.18	45.5158/9.1214	3.0	3.0	2.0	3.0
	45.5351/9.1609	3.0	3.0	2.0	2.0
	45.5023/9.1305	1.0	1.0	1.5	1.0
	45.4282/9.2459	1.0	3.0	1.0	1.0
–	–	–	–	–	–
	Average	1.59	1.90	1.75	1.54
	SD	0.83	0.87	0.66	0.68
	SE	–	0.58	0.51	0.46
	Unavailability (%)	0	0	0	0

Table 3 Vehicular traffic: excerpt of real traffic versus PAPT Estimations versus InfoBlu and GoogleTraffic (secondary roads in Milan)

Timestamp	Where (Lat/Long)	Real traffic	PAPT	InfoBlu	GoogleTraffic
10/06/2013 at 15.48	45.5172/9.2255	1.0	1.0	n.a.	2.0
–	–	–	–	–	–
	Average	1.34	1.28	n.a.	1.86
	Std deviation	0.41	0.52	n.a.	0.96
	Std error	–	0.35	n.a.	0.64
	Unavailability	0%	5%	100%	20%

Table 4 Speed reduction coefficients for speed profile

	Free (%)	Heavy (%)	Congested (%)	Impossible (%)
Daytime				
Residential/other	50.0	30.0	15.0	12.0
Tertiary/unclass	55.0	40.5	18.5	15.0
Secondary	60.0	50.0	20.5	15.5
Primary	70.0	55.5	25.2	15.5
Motorway	80.0	65.0	30.0	16.0
Night-time				
Residential/other	60.0	30.0	16.0	12.0
Tertiary/unclass	75.0	40.5	18.7	15.0
Secondary	80.0	50.0	20.5	15.5
Primary	84.0	55.5	25.2	15.5
Motorway	88.0	65.0	30.0	16.0

5 Speed profile estimations

5.1 Overview of the scenario

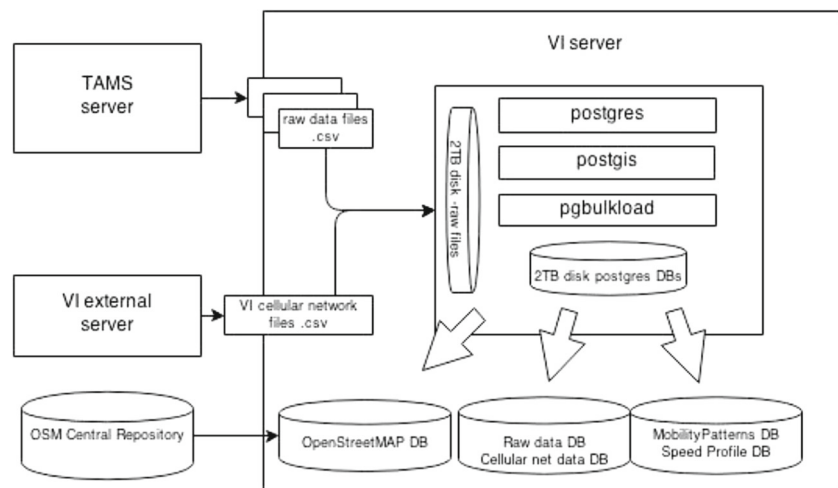
SpeedProfile estimations from the VI cellular network expose real time and historical average speeds for each OpenStreetMap OSM edge (i.e. road) of Milan (more than 300k edges covered).

The computation of SpeedProfile predictions starts from the observation of the traffic situations (see Sect. 4 for more details) computed from the VI cell net, from the maximum speed limit of each edge (as reported by OSM tags), and also

from a set of coefficients we defined to weigh each OSM edge in relation to the road type (as categorized by OSM tags) and the time of the day. The estimated Speed can be calculated as:

$$\text{estimatedSpeed} = \text{maxRoadSpeed} * \text{coeff} \quad (2)$$

where $0 < \text{coeff} \leq 1$ are calculated as detailed in Table 4. Figure 6 shows a Milan area with related speeds and traffic indications computed by the SpeedProfile service for each OSM road segment.

Fig. 5 VI Infrastructure for the SpeedProfile service

5.2 Technical aspects

SpeedProfile estimations refer to OSM road segments, that have been uploaded in a postgres/postgis DB, and that are referred to the OSM origin and end-points identifiers of the line representing the road segment. The OSM road segment identifiers are used by all sources to merge different speed predictions together. As a first step, we have to map the OSM road segments with the VI cellular network areas so as to use the traffic predictions during the speed profile calculation. For this purpose, postgis functions have been used to identify the VI network cell or cells that contain and cover the road segments. If a road segment is covered by more than one network cell, the cell that covers the biggest area of the road segment is the one associated with the road segment. A registry table has been created to store this mapping.

SpeedProfile estimations from VI cellular network are updated every 5 minutes by using the Postgres job scheduling agent *pgAgent*. SpeedProfile estimations are recorded in a dedicated VI database (see Fig. 5). The service can provide a completely update picture of all the 300k road segments with associated speed profiles.

The core of the algorithm is the way each OSM road type is weighed by a specific coefficient (computed both theoretically and also empirically) to provide differentiated average speeds for each road segment.

Table 4 summarizes the speed reduction coefficients for the SpeedProfile. The speed reduction factors depend on (1) the time period of the day: daytime or night-time; (2) the OSM road type classification: residential, tertiary, secondary, primary, motorway; (3) the traffic indicator: free flow, heavy, congested, impossible. The coefficients have been computed following the speed reduction ranges provided by DATEXII (<http://www.datex2.eu>), and they have been tuned overtime following the results of our experimental activities (see Sect. 3.3) Since these coefficients are an asset for VI,

Table 4 lists fake values that are not real ones used in the VI production environment.

Taking into consideration the maximum allowed speed per road types, which are stored into the OSM DB, can generate a set of possible combinations of speeds.

The current infrastructure for the SpeedProfile estimations from the VI cellular network implementation is illustrated in Fig. 4. In the diagram, we can see the three input data flows listed below:

1. From TAMS server, .csv files containing VI network events are pushed into the VI server, stored in a 2TB archive, and uploaded to RAW_DATA DB via the *pgbulkload* Postgres utility;
2. From VI external server, .shp files containing the VI network cellular configuration are provided and uploaded to the PostgreSQL/PostGIS Cellular Net data DB via the *pgbulkload* and the *shp2pgsql* utilities;
3. OSM geographical data extractions, including Milan map and road segments, loaded in the local Postgres/postgis OSM DB. Mobility Patterns and SpeedProfile DBs are in the same PostgreSQL DB Cluster, on VI server.

5.3 Experimental activities

This section explains the number of experiments we conducted in the VI production environment and with thousands of real VI mobile users to validate the SpeedProfile computation both from functional and non-functional perspectives.

From a functional point-of-view, we focused on understanding the quality of the SpeedProfile estimation as output of the designed algorithm and coefficients. From a non-functional perspective, we were interested in understanding whether the designed infrastructure is able to compute and update all the 300k OSM edges of Milan in real time. To this end, we set up the following real-world experiment: we

Table 5 SpeedProfile estimations versus GPS real data

	TRACK1	TRACK2	TRACK3	TRACK4	TRACK5	TRACK6
	21/02/2014	06/03/2014	14/03/2014	21/03/2014	21/03/2014	31/03/2014
	8.46AM	8.18AM	11.42AM	01.35PM	11.36PM	1.15PM
	Algorithm version 1.0			Algorithm Version 2.0		
GPS avg real speed (km/h)	18.2	26.0	42.5	22.0	59.8	29.6
SpeedProfile avg speed (km/h)	20.7	30.0	52.4	29.8	59.6	30.4
GPS total time (min)	57.0	55.0	6.7	35.0	10.35	40.3
SpeedProfile tot time (min)	42.4	48.0	5.4	26.2	10.09	39.3
Diff. GPS speed versus SpeedProfile speed (%)	+12.14	+13.45	+18.90	+26.18	−0.44	+2.53

asked VI colleagues to install the *myTracks* app for iPhone. This app allows tracking and saving .gpx files containing all the coordinates, lengths, real speeds, and elevations of a trip, in the form:

```
<trkpt lat="45.5539955339"lon="9.164810144339">
<ele>147.8531799316</ele>
<time>2014-02 26T07:46:28Z</time>
<extensions>
  <mytracks:speed>17.0645500668</mytracks:speed>
  <mytracks:length>0.0237007639816</mytracks:length>
</extensions>
</trkpt>
```

We selected a set of 6 tracks collected in February/March 2014 and in different daytimes, for a total distance of 100kms. We then compared the real GPS data with our SpeedProfile estimations to evaluate the percentage of error of our solution against real GPS data. Please note that coefficients have been fine-tuned overtime (the last two tracks use the current available coefficients.) As shown in Table 5, the SpeedProfile predictions overestimate the average speed in case of Track1, Track2, Track3, and Track4 (before the final tune of the coefficients as used in the current version of the algorithm V2.0.) In case of Track5 and Track6, the average trip speeds and times are more or less equal to the real data collected by the *myTracks* app. We are conducting extensive experiments to understand the quality of the predictions with additional journeys.

All the predictions have been computed in real time with the current version of the infrastructure, thus suggesting that the architectural and infrastructural choices we made are able to elaborate in real time the big data related to the SpeedProfile. Currently, we have also a running instance of the SpeedProfile service on a production environment, which is updated constantly—every 5 min—with the SpeedProfile estimations from the VI cell net and then used by a journey planner. The only problem experienced—and now solved—was related to disc space usage.

6 Origin/destination matrixes

6.1 Overview of the scenario

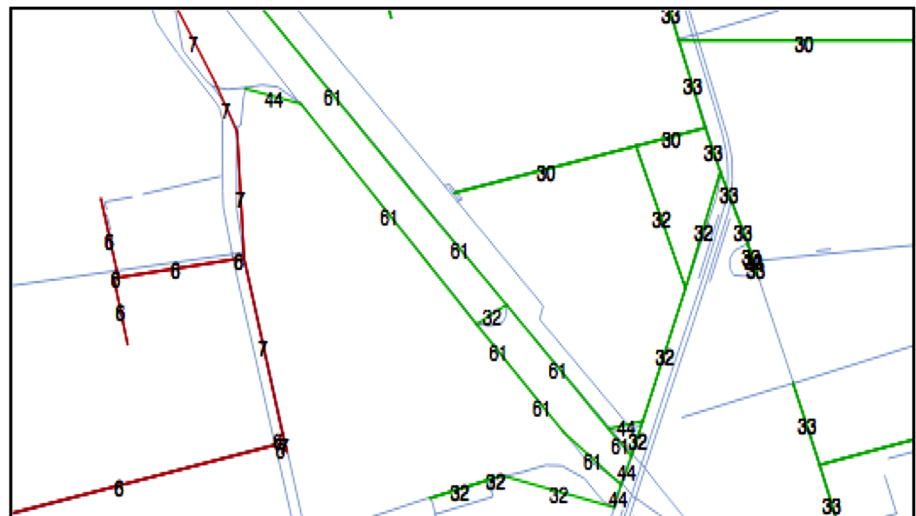
The Origin/Destination Matrix Mobility Patterns from VI cellular network describe the behaviour of people moving from an area i to an area j of Milan [11]. The city has been divided into several areas following the polygonal map drawn by the VI cellular network, for this purpose.

The two patterns provided for the O/D Matrix are the *Density* and the *Distribution Pattern*. The *Density Pattern* estimates the number of people moving from an origin area i to a final destination area j of Milan in one-hour time interval; the *Distribution Pattern* returns the list of densities of people moving from a Milan target area to a destination Milan area, for a one-daytime interval.

The algorithm to infer O/D Matrix exploits the tracking of mobile users (anonymized) to study the O/D flows of commuters and people in a city at real time. The algorithm provides the ability to track and analyse incoming and outgoing flows from the outskirts to the city centre and vice-versa or it facilitates the identification of the most congested roads in target hours of the day or of the week. For instance, this can be useful to plan logistics movements of hazardous goods.

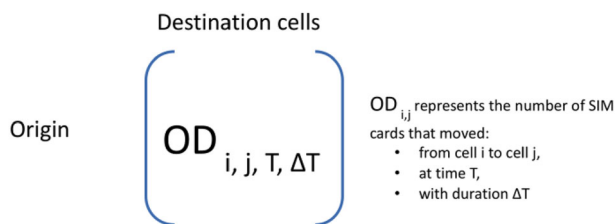
The pattern provides a view of O/D matrixes with a refresh time every 30 mins. The tuple $\langle i, j \rangle$ for the originating cell and the destination cell is stored in a dedicated DB, which contains the whole matrix of all possible combinations between origin and destination cells. Practically, for Milan, the DB stores 3000×3000 combinations of $\langle i, j \rangle$ cells. The diagonal of the matrix reports the number of mobile users that are considered by the algorithm as stationary (i.e. the mobile users that do not change their position in the last 5 mins of elaboration.) The implementation of the algorithm is parametric against all the discussed timeframes. Simplifying, the algorithm gets as inputs the following events:

Fig. 6 Milan area showing the real-time speeds computed by the SpeedProfile service



Event t : O -IMSI, cellID O , Timestamp, A ;
 Event $t + 1$: O -IMSI, cellID D , Timestamp, $A + x$.

These two events describe the movement from one point A (contained into the Originating Cell <CellID O >) to another point $A+x$ (contained into the Destination Cell <CellID D >) of a mobile phone described by its O -IMSI (i.e. an identifier derived by the IMSI through a crypto- algorithm in order to maintain the privacy of the mobile phone related to the collected network events.) Generalizing the discussion to all ΔT and to all cells of the network, the O/D matrixes are defined as:



6.2 Technical aspects

The elaboration of the O/D Matrix Mobility Patterns refers to the VI cellular network coverage areas and to one-hour timeframe, or a day timeframe, depending on the pattern type. The density estimations are scheduled at every hour of the day, using the Postgres job scheduling agent *pgAgent*, and recorded in the VI O/D Patterns DB.

The algorithm for elaborating the patterns makes use of the VI cells coverage information, and the O/D Matrix areas strictly depend on it. The algorithm computes the cellular network signalling events collected by the VI probe, sent by the TAMS server to the VI server and uploaded to the RAW_DATA DB, corresponding to the cellular network areas and to the related timeframe. The Patterns are com-

puted and then placed in the O/D Pattern DB and one-month history is maintained.

The *Distribution Pattern* stores the list of the calculated densities for the day and the related target and destination areas. All the signalling events generated by VI mobile users are read from the raw data table associated with the time interval to be observed. A count of unique users that move from an area i to an area j of the Milan map is then performed. The current VI infrastructure we set up for the O/D Matrix Mobility Patterns is the one already illustrated and discussed in Fig. 5. Note that, as in the case of the other patterns, the O/D Matrix Mobility Patterns only apply to Milan city trial, as the Mobility Patterns provided are built up based on traffic events generated in the Vodafone Italy cellular network, collected by the probe and transferred by the TAMS server to the VI server (Fig. 6).

The raw data files are an input source to the Mobility Patterns infrastructure, together with the cellular network input files that provide the Vodafone cellular coverage information, needed to map the events to the areas of the city, accordingly. These files are uploaded into the *postgres* databases using the *pgbulkload* utility. Mobility Patterns are then valued and stored in the related MobilityPatterns DB.

6.3 Experimental activities

Observations and extractions of the O/D Matrix flows Patterns have been performed; no strong validations have been performed for this algorithm, since the most updated census data, related to Milan, are back to year 2006. Hence, here we only show some graphical representations of the O/D matrix data. In Fig. 7, a central area of Milan is highlighted together with morning incoming flows during a one-hour observation (green arrows), and outgoing flows from the same area (red arrows), for an evening hour. Areas with higher densities have



Fig. 7 Incoming/outgoing flows (morning and evening)

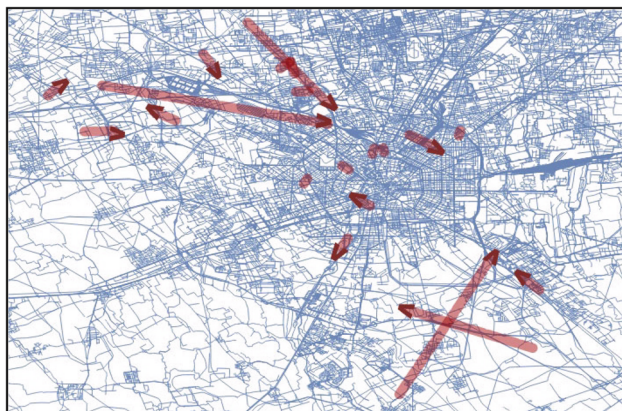


Fig. 8 Milan area with most relevant commuters' flows (morning)

been observed separately, as the complete picture of Milan results in a very dense diversified and overlying O/D flows. In Fig. 8, we show the most congested Origins and Destinations of Milan. The picture refers to the day May 27th at 8.00AM and highlights the O/D trips and directions in this rush hour, showing how commuters move in the morning.

Additional analysis can be further investigated by exploiting the concepts introduced in the Spatial Interaction Model approaches [20]. Heterogeneous areas of a city can be grouped to determine a limited number of locations in geographical space and thus to observe specific behaviours and specific phenomena that are subjected to dependency and heterogeneity. For example, in the context of O/D matrixes, policy makers can be interested in understanding if indigent outskirts of the city have an impact on traffic congestions more or less than other areas of the city, or to understand how these areas use public transportations. Moreover, geographical and behavioural data can be enriched with third-party data (e.g. census data) to create heterogeneous spatial profiles and new scenarios of data elaboration.

It is important to highlight that the geometry of these geographical spaces is fixed, because we use the network topology of the Telco operator as starting point of movement observations.

7 POI mobility study

7.1 Overview of the scenario

The point-of-interests (POIs) Mobility pattern estimates the number of people stationing/moving around a set of target Milan POIs, in a certain time interval. The list of POIs includes Milan architectural buildings, museums, theatres, and main railway stations. We selected eleven POIs of Milan, as working examples. In any case, this list can be updated and extended with new POIs, by simply adding new POIs in a dedicated .csv file.

The available types of POI Mobility Patterns are the *Density POI Pattern* and the *Distribution POI Pattern*, which estimate the number of people situated nearby a POI in a target time interval and also the daily distribution of densities of people in the POI areas, respectively. The densities are related to one-hour time intervals. The densities are calculated as soon as the data acquisition from the Vodafone cellular network—for the target time interval—completes. It has been decided to set the time interval for densities calculation to one-hour, but there is no restriction for providing the patterns for smaller/bigger time intervals.

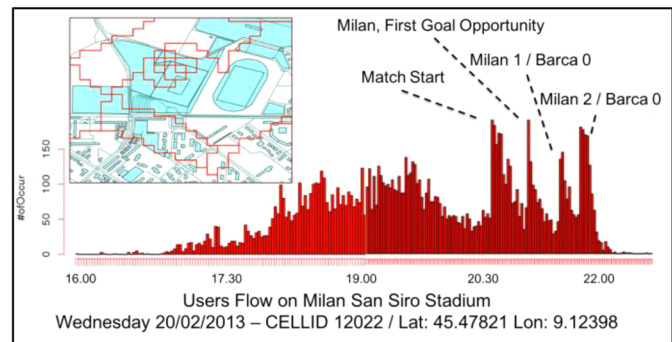
Figure 9 shows the selected POIs, their geo-locations in the map of Milan, and an example of the real-time density of each POI (proportionally to the bubbles dimension.)

For instance, we used the POI mobility pattern to observe in real time the behaviour of people during big events, such as how people reach stadium or arena for football matches or concerts. This is important for policy makers to understand how to avoid congestions in the city and how to act on Public Transportation to rule people flow. As an example, we observed the behaviour of people during the football big match Milan versus Barcelona (Championship match 20 Feb. 2013—75.000 total spectators). Figure 10a shows a graphical representation about how people in the San Siro Stadium interact with their smartphones. It is interesting to observe that the graph shows four peaks (e.g. when the match started; during the first opportunity of goal by Milan; at the first goal

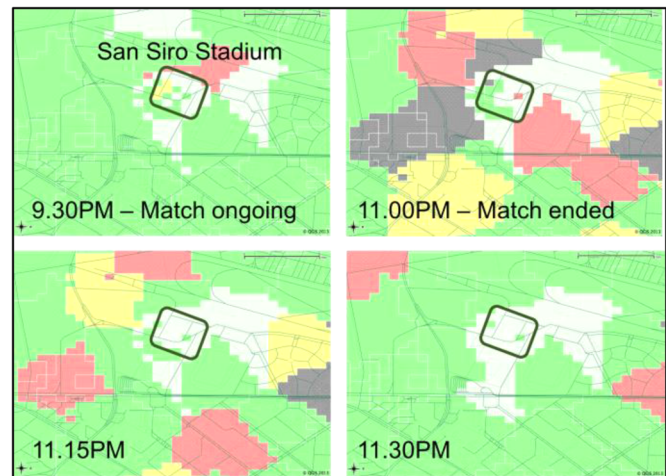


Fig. 9 Milan POIs selected for the POI service

Fig. 10 Champions big match, San Siro Stadium—Milan versus Barcelona



(a)



(b)

by Milan; at the second goal by Milan), thus indicating that people strongly interact with their smartphones (for example, to post photographs of the stadium, and update their social network profiles.).

Figure 10b shows the people flow when the match ended. During the match, the people are exclusively in the stadium (9.30PM—red area). At the end of the match, people start exiting the stadium, thus overcrowding the areas around San Siro (11.00PM). The overcrowded areas overtime move away from the stadium (11.15PM), and the overcrowded situation around the stadium is completely free at 11.30PM.

7.2 Technical aspects

The infrastructure for the POIs pattern is the one depicted in Fig. 5. The POIs basic information is saved in the POI Pattern DB registry, where also network cells coverage information is stored. The size of the geographical area covering the POI is taken into consideration. Consequently, the number of people observed in the area strictly depends on the VI cellular network topology and on the layout of the cell or cells that intersect with the geographical area covering the POI. Hence, the first step to the definition of the POI Pattern is to create the registry of the POIs. This can be done by the following

algorithm that (1) extracts from a .csv file all target POIs to be considered, (2) creates the geometry for each POI, and (3) relates each POI geometry with the VI cell/cells that cover each POI.

The core of the POI Pattern is the algorithm to estimate people densities around POIs. The algorithm takes in input the raw data coming from the TAMS server to return the total count of people moving around POIs. The algorithm (1) extracts, for each POI, the cells that intersect with the geographical area covering the POI; (2) counts the distinct Obfuscated-IMSI (O-IMSI) present in these cells in the selected time interval; (3) updates the POI DB table with the result of the count.

People densities are provided to external third-party journey planners or policy makers after multiplying the calculated values by a multiplication factor, which we derived from the analysis of the 3G penetration and the Vodafone Italy market share, and by averaging these values with empirical values collected during big events in the city. To this end, we observed the ratio between cellular network distinct events and the real number of people during football matches in the San Siro Stadium. Averaging the 3G penetration, the “phone on and with me” percentage as reported in [8] and the VI market share, we obtained a multiplication factor equals to

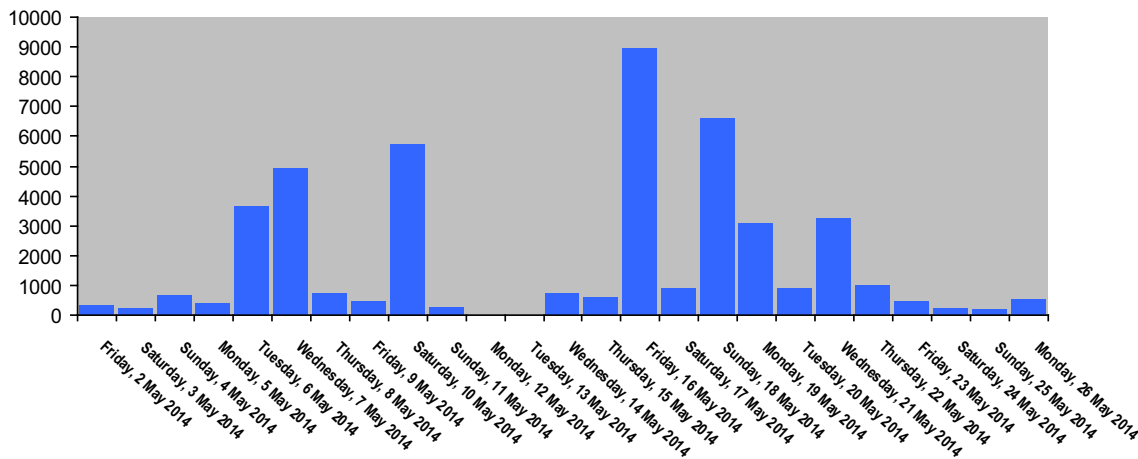


Fig. 11 Milan Assago Forum crowd estimations (May2014) 9.00PM)

11.9. The empirical part, with 26 data points and referring only to VI primary cells, suggested a factor equals to 13.6. The final average value we used to count people is: **12.7**. When considering also VI secondary and tertiary network cells, the average value decreases to **9.5**. We use this factor to multiply the number of people detected by the algorithm to have people count estimations close to the reality. This coefficient is applied to all Mobility Patterns generated by the VI cellular network.

7.3 Experimental activities

We validated the POI patterns calculated from real data collected from the VI network by means of the k -fold cross-validation technique [6] to understand the quality of our people estimations. In k -fold cross-validation, our original sample composed of 26 data points, is randomly partitioned into k equal size subsamples. Of the k subsamples, a single subsample is retained as the validation data for testing the model, and the remaining $k - 1$ subsamples are used as training data. The cross-validation process is then repeated k times (the *folds*), with each of the k subsamples used exactly once as the validation data. We created 5 folds (each with 5 randomly selected data points), and we used the Standard Error SE (see Formula 3) to evaluate the quality of each fold and the relative average quality of our people count (one outlier has been discarded in the analysis.) The k -fold average SE is equal to 5489 and the relative RSE is 13.8%. Since the RSE is less than 25%, the people count estimation can be considered reliable enough for general adoption.

As for the confidence intervals for the mean of people counted in this cross-validation experiment:

$$\text{Upper 95\% limit} = 34.524 + (5489 * 1.96) = 45.282$$

$$\text{Lower 95\% limit} = 34.524 - (5489 * 1.96) = 23.766$$

where 34.524 is the avg predicted value and 1.96 is the 0.975 quantile of the normal distribution. It can, therefore, be considered with 95% reliability that the true value of people count is between 23.766 and 45.282. In this case, the avg real count is: 39.769.

Moreover, we used new additional data points to validate the estimations of the POI pattern. In Fig. 11, we compare people count estimations in the area of the Milan Assago Forum (May 2014 at 9.00PM). Data related to days 12th and 13th of May are not available due to maintenance activities of the VI infrastructure. Higher values correspond to concerts or big events that took place at the Assago POI: the 6th and 7th of May Baglioni's concert, the 10th of May Giorgia's concert, 16th and 18th of May Final for Basket Euroleague, 21th of May EA7 play-off basket.

Another observation (see Fig. 12) relates to the RadioItalia live concert that took place on June 1st 2014 in the Milan Cathedral area. In the figure, we can see a comparison of crowd densities in the Milan Cathedral area on a day where a very popular concert takes place, and days before and after, where no event takes place. The density pattern calculated for the area in the timeframe shows 91.000 people stationing at around the Milan Cathedral. This estimation is in line with the number of participants to the concerts, as reported by the press article in the newspaper: Corriere.it (<http://milano.corriere.it>).

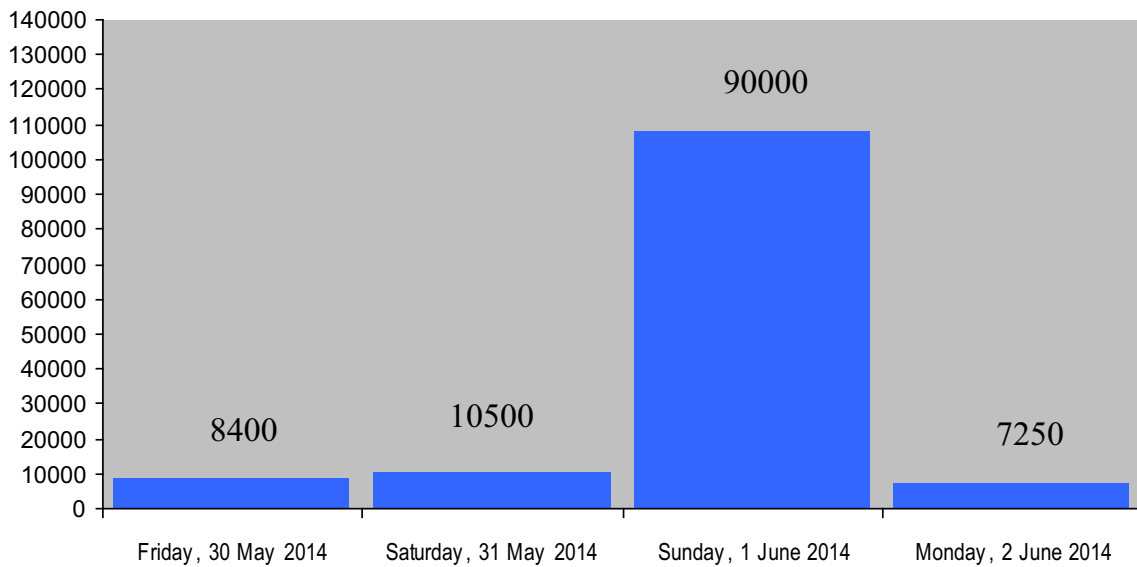


Fig. 12 Milan Cathedral crowd estimations (June 1st 2014, 9.00PM)

Summarizing, our approach estimated:

Event	People count (estimated)	People count (real)	Relative error (%)
Baglio ni’s concert, May 6th	4000	Na	Na
Baglio ni’s concert, May 7th	5000	Na	Na
Giorgi a’s concert, May 10th	6000	Na	Na
Eurole ague, May 16th, 18th	11000	12300	−10.0
EA7 Pl ay-off, May 21st	5400	6700	−18.5
RadioI talia, June 1st	91000	100000	−9.0
Cyrus’ s concert, June 8th	10500	10000	+5.0
Ligabu e’s concert, June 6th	66600	65000	+2.5
EA7 versus Siena, June 17th	11000	12100	−9.0
Pearl Jam concert, June 20th	64000	62000	+3.0
		AVG	−5.1%

Both the k-fold cross-validation and the validation against new big events in the city show the potentiality of the algorithm. The k-fold RSE is equal to 13.8% (less than the theoretical max threshold of 25%). Moreover, the real validation of the algorithm show an increase in precision compared to the RSE: the average RE for the seven big events analysed is equal to −5.1%, suggesting that the algorithm underestimate in average a little bit.

8 Subway flows mobility study

8.1 Overview of the scenario

The Subway Flows Mobility Pattern aims at describing the flows of people entering, moving through stations, and exiting the subway network, in particular the three main lines of the Milan subway network: the red (M1), the green (M2), and the yellow one (M3).

The Subway Flows Pattern provides the journey planners and Policy Frameworks with two main patterns: The first one returns real-time density situations where the estimated number of people in a subway station (or line) is computed by observing the behaviour of VI mobile users in a five minutes time-window; The second one returns historical distributions of people where the estimated number of people is computed for a one-day interval. See Fig. 13 for a graphical representation of the Subway Mobility Pattern output where numbers are the real-time density of users (for each station) computed by the pattern.

8.2 Technical aspects

To support the use of mobile phones in the Milan underground, VI covered the indoor area of the three Milan subway lines with 2G/3G micro-cells. A single micro-cell, in the VI subway network configuration, covers two, three or even four adjacent subway stations. Hence, the cellular data events, which are collected by the VI probes, are related to 3G micro-cells that do not have a one-to-one relation with subway stations. We denominate a cluster of stations the set of subway stations covered by a unique micro-cell. This

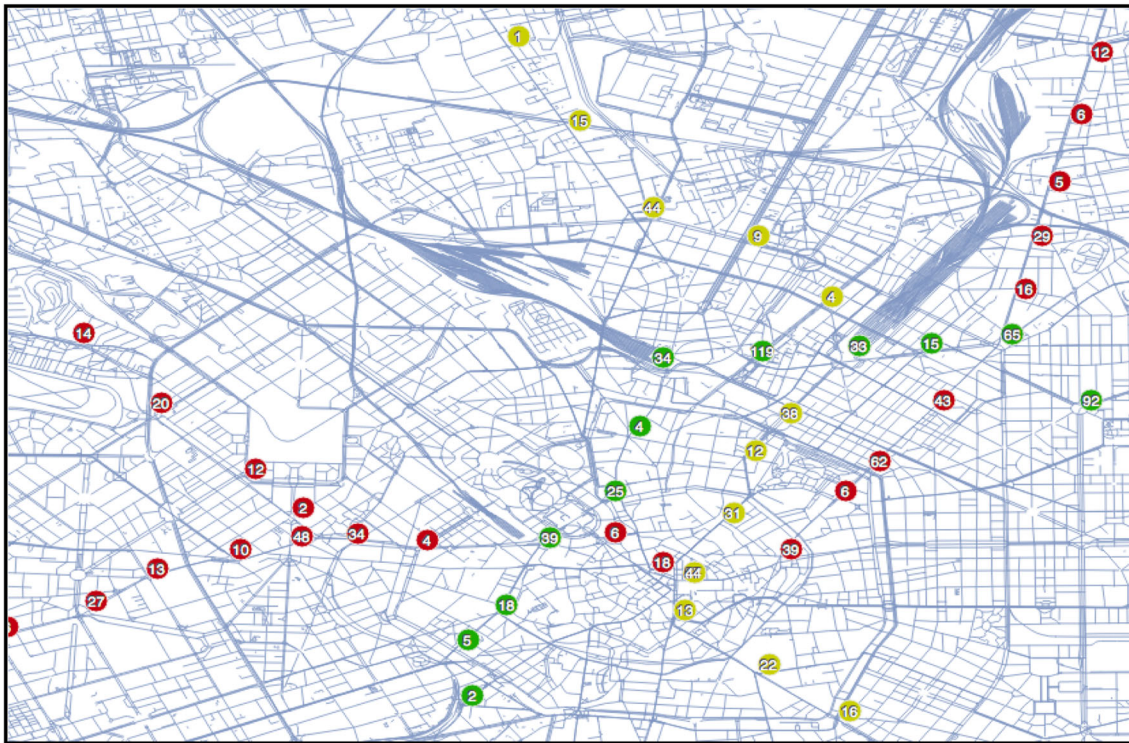


Fig. 13 Real-time density of Milan subway passengers

strongly complicates the models that we defined to describe the density and distribution patterns with the single station granularity.

Our defined models treat mathematically the subway lines as Finite State Machines (FSMs) where each station is a *state* of the FSM. The connections between stations are modelled as *transitions* from one state to the other one. Signalling events generated by the VI cellular network have collected and seen as *weights* for each transition. People flow entering and exiting subway stations is measured by coupling and counting two events of the same user, in a very short timeframe, related to a subway cluster. Actions of entering and exiting subway stations are related to ground level cells that cover each subway station entrance. To identify the cells that cover subway entrances, and then be able to calculate people entering and leaving the subway stations, we analysed a one-day set of data, where we observed for very short time intervals (two/three minutes), the users that move from a network cell to a subway micro-cell, and vice-versa. Doing so, we identify those cells covering stations entrances, and, depending on the direction, we can derive if people are entering or leaving the stations.

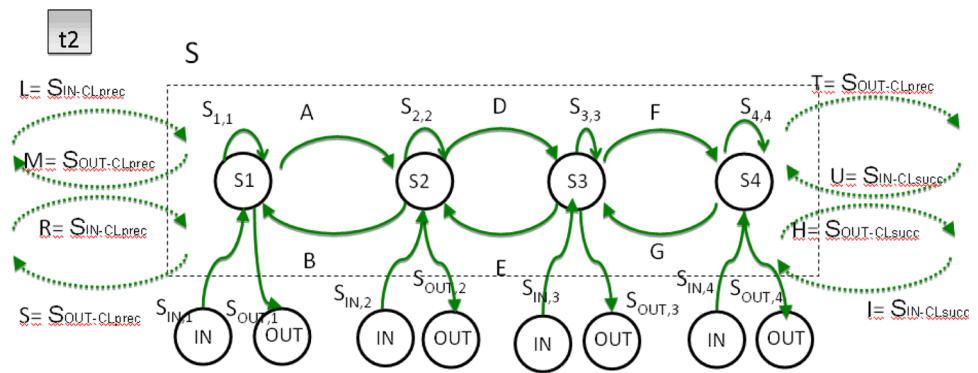
Weights for the FSM transitions are determined by coupling two events, in a very short timeframe, related to adjacent subway clusters and to the same user identified by its O-IMSI, where the direction is determined by the timestamp of the generated event. In other words, each time two events for

the same user in a short time period and at adjacent clusters are detected, a person moving from one station to another is counted; by counting the number of these detections, the system is able to calculate the weight for each station-to-station transition of the FSM.

The mathematical model we derived is depicted in Fig. 14. The model describes the case of subway clusters composed of four subway stations (i.e. each VI micro-cell covers four subway stations) but it is generalizable also in the case of subway clusters composed of 2 or 3 stations.

The mathematical model can be defined based upon all the events related to t_0 , t_1 and t_2 , and then starting the calculation of people displacements. Timeslots have been defined for this purpose, where t_0 corresponds to the opening time of the Milan subway stations, and subsequent intervals of five minutes correspond to the successive timeslots, till the time closure of the Milan subway, at the end of the day. We can detect and sum up the events generated by the cellular network, which are associated with a target subway cluster, analysing the cell-id reported in the generated event. We started considering the scenario at time t_0 , when subway stations open: At time t_0 , we expect that only inputs (IN) to stations (from subway physical station entrances) take place, because of at t_0 , neither exits from the subway stations nor displacement between adjacent stations can occur. At time t_1 , we assume that people who entered the subway stations at t_0 start moving to the next station of the same cluster (or to

Fig. 14 FSM representing a cluster of four stations



another cluster), and that new inputs to stations (from station entrances) occur. L, R, U, I identify inputs to the border stations of the cluster from adjacent stations that do not belong to the same cluster. A cluster CL might have, or not, one or two adjacent clusters, depending on the subway network configuration. So L, R, U, I are optional inputs. Similarly, M, S, T, H are optional outputs to adjacent stations in a subsequent cluster CL . A, B, D, E, F, G identify the displacements (transitions) between adjacent stations in the cluster, and the displacements between adjacent stations that belong to different clusters. At $t1$, outputs are considered as forbidden. At time $t2$, we have people entering, exiting and moving across the whole subway network. Displacements between interchange stations are to be considered as IN/OUT of the interchange stations involved. We can generalize the model to an instant t_i and we can mathematically describe each station and cluster CL as follows:

factor for each station in a cluster to weight the total number of distinct $O-IMSI$ s detected for the cluster. The corrective factor is then used to proportionally divide the total number of distinct $O-IMSI$ s among the different stations of the cluster in a specific timeframe. This is an improvement we applied to the algorithm to mitigate the issue related to micro-cells that cover several stations. The corrective factor and the weight of each station is dynamically computed by counting the number of people entering and exiting the target station in a target timeframe, instead of simply dividing the total number of detected passengers for a cluster by the number of stations per cluster. Further, as the data collected from the network refer only to events generated by the 3G VI network, the people density estimation takes into account the corrective factor that compensates the 3G penetration and the VI market share, as explained at the end of Sect. 3.2.

From an infrastructural point-of-view, the algorithm makes use of dedicated registries to store data coming from subway

$$\begin{cases} S1_{ti} = S1_{ti-1} + S_{INti,1} - S_{OUTti,1} - S_{INti-1,1} + S_{OUTti+1,1} \\ S2_{ti} = S2_{ti-1} + S_{INti,2} - S_{OUTti,2} + S_{INti-1,1} - S_{INti-1,2} + S_{INti-1,3} - S_{OUTti+1,4} - M_{ti} - S_{ti} + I_{ti} + U_{ti} \\ S3_{ti} = S3_{ti-1} + S_{INti,3} + S_{INti-1,2} - S_{INti-1,3} + S_{INti-1,4} - S_{OUTti,3} - S_{OUTti+1,1} + L_{ti} + R_{ti} - T_{ti} - H_{ti} \\ S4_{ti} = S4_{ti-1} + S_{INti,4} - S_{OUTti,4} - S_{INti-1,4} + S_{OUTti+1,4} \\ S_{ti} = S1_{ti} + S2_{ti} + S3_{ti} + S4_{ti} = CL \end{cases}$$

where S_n, S_{INn} and S_{OUTn} , are the outputs of the algorithm and they correspond to the flows of people entering, moving, and exiting the subway cluster and network.

Algorithmically, the *Subway Density and Distribution Patterns* is based on both offline and online phases. During the offline phase, all the data related to the subway network are managed to create a set of registries with the lattice of the network and all related information. During the online phase, the VI cellular network events are elaborated and used at real time to exercise the subway mathematical model and to estimate density and distribution values of people moving around the Milan subway.

To this end, we collect in/out displacements and transitions among stations. Moreover, we defined an adaptive corrective

stations and clusters (micro-cells) and ground level data (normal cells) that cover station entrances. Currently, the Subway Mobility Pattern has been released as a service to external third-party Journey Planners or Policy Frameworks. The infrastructure for the subway Flows pattern is common to the VI infrastructure in Fig. 5.

8.3 Experimental activities

We simulated the algorithm in a controlled environment to detect possible anomalies and misbehaviour in the mathematical model, and then we also experimented the implementation of the model and algorithms on real data sets collected from the VI production network to detect func-

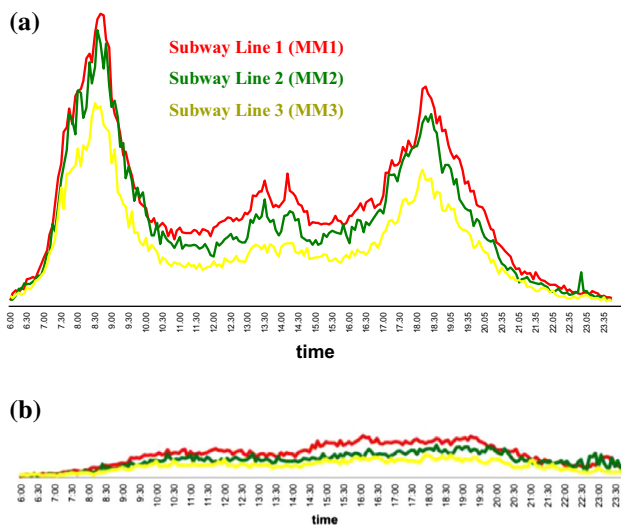


Fig. 15 Milan subway lines 1, 2, 3 daily distribution of users on Monday May 19th (a) and Sunday May 18th (b)

tional and non-functional limits in the algorithm and in its implementation. The simulations suggested us that the use of the corrective factors to weight each station is fundamental to have a real balance among stations, overtime. Exercising the subway pattern algorithms in a real context, we analysed the daily distribution of users in the three main Milan subway network lines.

In Fig. 15, we can observe the differences of a daily distribution of users during a working day (Monday May 19th 2014) and during a non-working day (Sunday May 18th 2014) computed by our service. As expected, in the first case, the highest concentration of people take place during the early morning hours and evening hours, and are related to journeys to/from work. In the second case, the daily distribution of users slowly grows in the morning, to slowly decrease in the evening.

To validate the output of the model and algorithms, we asked for official statistics. Unfortunately, these data are not available. Hence, we surfed the web to find statistics related to how Milan citizens use the three subway lines. No up-to-date and precise data are available. We found aggregated data only that refer to year 2012 and that show an average daily ridership equals to 1.15M users (www.atm.it). Our model counted for the whole day (Monday May 19th 2014) a total of daily ridership: 1.43M round trips (e.g. in this case, each round trip of a single passenger is counted by the model.) The count for a non-working day (Sunday May 18th) returned 480k round trips. The average daily value counted in this target week (21st week of the year) is: 1.21M round trips. Of course, the two data sets are not comparable for several reasons: They refer to different periods (2012 for the official ATM-MI data, 2014 for our counts). ATM-MI is the Milan Public Transport Company (www.atm.it/en). We are not able

to find details on the ATM-MI statistics, such as if the total number is computed by counting the total tickets sell or by counting the accesses to the turnstiles, or by means of other estimations.

9 Threats to validity

A number of threats may exist to the validity of a correlational study like ours. We now examine some of the most relevant ones.

9.1 Internal validity

We checked whether variables are normally distributed when carrying out regressions, as required by the theory of regression. Consistent with the literature, we used a 0.01 statistical significance threshold, the same we used for all statistical tests in our paper. The vast majority of statistical tests we carried out to this end provided quite strong evidence that the variables are indeed normally distributed for all the evaluated cells. These values are close to the 0.01 statistical significance thresholds, but based on the other indicators the models are not relevant. At any rate, the statistical tests used in regression are somewhat robust and they can be practically used even when the variables' distributions are not that close to normal.

9.2 External validity

Like with any other correlational study, the threats to the external validity of our study need to be identified and assessed. The most important issue is about the fact that our six selected cells cover Milan areas that are different each other (this is also highlighted by the different obtained regression results). In any case, this may have not somewhat influenced the results.

It was not possible to formally understand why in some cases the models are relevant and in other cases this is not true. This is due to the impossibility to access the algorithms used by the InfoBlu service, thus making hard to understand the reliability of their predictions. It is quite clear that in case of areas that cover primary roads, the InfoBlu predictions are more accurate because based on inductive loops and a lot of GPS data (this is the case for instance of cells 40972, 63601) while in case of small roads the predictions are less precise because probably based only on small GPS datasets. For example, we observed that for the area covered by the cell 40361 (Viale Cirene, Milan) the InfoBlu traffic prediction is in average equal to 4.8 (impossible traffic) and this is not in line with the real traffic scenarios for that area.

Moreover, the interpretation of the actual traffic status, observed by manually analysing the Autostrade, InfoBlu,

GoogleTraffic diagrams, and the Milan Webcam, is open to subjectivity. In any case, this does not strongly influence the computation of the standard error because the subjectivity errors propagate both when computing the standard error for all the pairs in our experiments.

9.3 Construct validity

An additional threat concerns the fact that the measures used to quantify the relevant factors may not be adequate. This paper deals with the number of network events collected by the probe and not filtered and elaborated by the algorithms before their use to compute the models. It is clear that the algorithms can improve the quality of the models and the reliability of the traffic predictions by removing “noising” events (note that these events marginally impact on the total number of collected events, thus inducing probably a small quality improvement of the model). The derivation of new regression models based on filtered data is for future work.

10 Conclusion

In this paper, several big data mining algorithms have been discussed to support the prediction and estimation of vehicular traffic conditions, speed profiles for roads, flows of people moving among subway stations and around POIs of the city, and also O/D matrices, for future smart cities. The paper provided the reader with a general description of the features and their technical aspects to support real-time elaboration of big data coming from the VI cellular network. All the features and mobility patterns have been deeply experimented in real-life situations and in the VI production environment with thousands of real VI mobile users (anonymized). Where possible, the estimations have been also validated against real and official data. The results of the experimental part are very promising. The quality of the predictions and estimations is often better than well-adopted competitors and in most cases, the designed mining algorithms fill the gap of still missing solutions both from a research as well as from the industrial point-of-view. Solutions to model subways flows are unavailable in the literature. This is also true in the industrial setting: ATM-MI does not have a solution to count people moving among subway stations in real time. This can be considered as a real added value for future smart cities.

References

- Tosi, D., Marzorati, S.: Big data from cellular networks: real mobility scenarios for future smart cities. In: Proceedings of the IEEE Big Data Service Conference, Oxford, March 2016
- Tosi, D., LaRosa, M., Marzorati, S., Dondossola, G., Terruggia, R.: Big data from cellular networks: how to estimate energy demand at real-time. In: Proceedings of the IEEE International Conference on Data Science and Advanced Analytics (DSAA 2015), Paris, Oct 2015
- Modoni, G., Tosi, D.: Correlation of weather and moods of the Italy residents through an analysis of their tweets. In: Proceedings of the Third IEEE International Symposium on Social Networks Analysis, Management and Security (SNAMS), Vienna, 2016
- González, M.C., Hidalgo, C.A., Barabási, A.-L.: Understanding individual human mobility patterns. *Nature* **453**, 779–782 (2008)
- Tosi, D., Marzorati, S., Pulvirenti, C.: Vehicular traffic predictions from cellular network data—A real world case study. In: Proceedings of the IEEE International Conference on Connected Vehicles and Expo (ICCVEx), Vienna, 2014
- Fisher, R.A.: On the interpretation of χ^2 from contingency tables, and the calculation of P. *J. R. Stat. Soc.* **85**(1), 87–94 (1922)
- Geisser S.: Predictive Inference. New York: Chapman and Hall. ISBN 0-412-03471-9. (1993)
- Webcam Viale Monza. Web Published. Accessed Dec 2014. www.webcam-4insiders.com/it/meteo-Milano/5258-Milano-meteo.php (2014)
- Our Mobile Planet—Italy. Identikit dell’Utente Smartphone. Google Report Accessed Dec 2014. <http://services.google.com/fh/files/misc/omp-2013-it-local.pdf> (2014)
- OpenStreetMap Road Classification. Web Published. Accessed Dec 2014. <http://wiki.openstreetmap.org/> (2014)
- Fiadino, P., Valerio, D., Ricciato, F., Hummel, K.: Steps towards the extraction of vehicular mobility patterns from 3G signaling data. In: Proceedings of The 4th International Workshop On Traffic Monitoring and Analysis, (TMA’12), Vienna, March 12, 2012
- Calabrese, F., Di Lorenzo, G., Liu, L., Ratti, C.: Estimating origin-destination flows using mobile phone location data. *IEEE Pervasive Comput.* **10**(4), 36–44 (2011)
- Li Mei, G., Da Yong, L.: Apply cellular wireless location technologies to traffic information gathering. In: Proceedings of The 2nd International Conference on Intelligent Computation Technology and Automation (ICICTA’09), pp. 499–502, (2009)
- Valerio, D., Witek, T., Ricciato, F., Pilz, R., Wiedermann, W.: Road traffic estimation from cellular network monitoring: a hands-on investigation. In: Proceedings of the 20th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC’09), pp. 3035–3039, (2009)
- Valerio, D., D’alconzo, A., Ricciato, F., Wiedermann, W.: Exploiting cellular network for road traffic estimation: a survey and a research roadmap. In: Proceedings of the 60th IEEE International Conference on Vehicular Technology (VTC’09), pp. 1–5, (2009)
- Shashikiran, V., Sampath Kumar, T.T., Sathish Kumar, N., Venkateswaran, V., Balaji, S.: Dynamic road traffic management based on Krushkal’s algorithm. In: Proceedings of the IEEE International Conference on Recent Trends in Information Technology (ICRTIT’11), pp. 200–204, (2011)
- Calabrese, F., Colonna, M., Lovisolo, P., Parata, D., Ratti, C.: Real-time urban monitoring using cell phones: a case study in Rome. In: IEEE Transactions on Intelligent Transportation Systems (TITS’11), (2011)
- Di Lorenzo, G., Luca Sbodio, M., Calabrese, F., Berlingiero, M., Pinelli, F., Nair, R.: AllAboard: visual exploration of cellphone mobility data to optimise public transport. *IEEE Trans. Vis. Comput. Graph.* **22**(2), 1036–1050 (2016)
- Sohn, T., Varshavsky, A., LaMarca, A., Chen, M. Y., Choudhury, T., Smith, I., Consolvo, S., Hightower, J., Griswold, W. G., de Lara, E.: Mobility detection using everyday GSM traces. In: Proceedings of the 8th International Conference on Ubiquitous Computing (UbiComp), pp. 212–224, (2006)
- Upton, J.G., Fingleton, B.: Spatial Data Analysis by Example Volume 1: Point Pattern and Quantitative Data. Wiley, New York (1985)

21. Remy, J.: Computing travel time-estimates from GSM signalling messages: the STRIP project. In: IEEE Intelligent Transportation Systems Conference, Oakland (2001)
22. Regression and Least Squares Fitting. Web published. Accessed Dec 2014. <http://mathworld.wolfram.com/LeastSquaresFitting.html> (2014)
23. Herrera, J.C.: Evaluation of traffic data obtained via GPS-enabled mobile phones: the Mobile Century field experiment. *Transp. Res. C Emerg. Technol.* 18(4), 568–583 (2010)
24. Herrera, J.C., Work, D.B., Herring, R., Ban, X., Bayen, A.M.: Evaluation of traffic data obtained via GPS-enabled mobile phones: the mobile century field experiment. In: Proceedings of ACM Mobisys, (2009)
25. Gonzalez, H., Han, J., Li, X., Myslinska, M., Sondag, J.P.: Adaptive fastest path computation on a road network: a traffic mining approach. In Proceedings of VLDB, (2007)
26. University of Maryland Transportation Studies Center. Final Evaluation Report for the CAPITAL-ITS Operational Test and Demonstration Program. TR, 1997