An Unsupervised and Non-Invasive Model for Predicting Network Resource Demands

Fulvio Corno, *Member, IEEE,* Luigi De Russis, *Member, IEEE,* Andrea Marcelli, *Student Member, IEEE,* Teodoro Montanaro, *Student Member, IEEE,*

***Abstract*—During the last decade, network providers are faced by a growing problem regarding the distribution of bandwidth and computing resources. Recently, the mobile edge computing paradigm was proposed as a possible solution, mainly in consideration of the provided possibility of transferring service demands at the edge of the network. This solution heavily relies on the dynamic allocation of resources, depending on the user needs and network connection, therefore it becomes essential to correctly predict user movements and activities. This paper proposes an unsupervised methodology to define meaningful user locations from non-invasive user information, captured by the user terminal with no computing or battery overhead. The data is analyzed through a conjoined clustering algorithm to build a stochastic Markov chain to predict the users' movements and their bandwidth demands. Such a model could be used by network operators to optimize network resources allocation. To evaluate the proposed methodology, we tested it on one of the largest public community's labeled mobile and sensor dataset, developed by the "CrowdSignals.io" initiative, and we present positive and promising results concerning the prediction capabilities of the model.**

***Index Terms*—Unsupervised modeling, Markov chain, network resources allocation, non-invasive user information**

I. Introduction

The rapid growth of connected devices and online services, one of the effects brought by the increase of the adoption of Internet of Things (IoT) in the human society, introduces new challenges for network providers. In fact, as reported by Brogi *et al.* [1], it is extremely difficult to support the transfer of data from billions of IoT devices due to the volume and the geo-distribution of those devices. In addition, the need of reduced latency, high quality connections, and the availability of storage closer to where data is generated, is evident.

Recently, the mobile edge computing (MEC) paradigm was proposed as a possible solution to such needs and problems: it consists of a strongly virtualized platform that delivers a rich portfolio of services and applications at the edge of the network [2], [3]. To the basics, the MEC architecture defines heterogeneous intelligent nodes, called Edge Data Centers (EDC), which are mainly distributed at the proximity of the users. However, a high user mobility generates a new issue in the management of the resources provided by the EDCs. In fact, as users move across different geographical areas, they should be ideally connected to the closest EDC and this causes large variations in resource demand at each EDC [4]. Thus, the load required to an EDC could be very intense in some moments (e.g., during an event, like a concert) and very low in others, and such a variability should be addressed by the network providers during the design of their networks, while both reducing costs and enhancing user satisfaction.

One possible solution, already described in the literature (e.g., [4]–[6]), is to predict the future resource demand in each EDC to efficiently and dynamically adjust the EDC capacity while maximizing resource utilization.

These works apply the capability of estimating user meaningful locations (i.e., locations that carry some meaning to the user and to which the user can potentially attach some meaningful semantics [7]) to the maximization of global network resources, in order to dynamically adapt EDC capacity to day time and other user constraints.

In this paper, we extend the original concept of user meaningful locations, and we define the user *meaningful network locations* as user specific recurring spatio-temporal conditions, spatially identified by the user connected networks, and where the user spends a significant portion of time (e.g., more than one hour). Each location can be later assigned a network usage pattern that can be exploited by network provides to predict and dynamically adapt the EDC capacity to day time and other user constraints. Indeed, human mobility is highly regularized rather than randomized in both temporal and spatial domains [8], and if it is possible to predict the future network bandwidth demand, based on user mobility, it is possible to optimize the resources allocation too [9].

This paper introduces an unsupervised methodology to support network providers in predicting user meaningful network locations, based on *non-invasive* users information, like Wi-Fi- and cellular-based inferred locations. Non-invasive users information is the kind of data that can be accessed by network providers without any additional license agreements, and users are already used to its collection, as mobile operators and device manufacturers commonly acquire it in order to improve the quality of their services. Compared to GPS location, network inferred location is less invasive for the user privacy, it has a negligible impact on the battery consumption, and it provides the level of detail required to predict the resource demand in each ECD. To the best of our knowledge, this is the first work that uses non-invasive user information, collected

F. Corno, L. De Russis, A. Marcelli and T. Montanaro are with the Department of Control and Computer Engineering, Politecnico di Torino, Corso Duca degli Abruzzi, 24 - Torino, Italy 10129 e-mail: {fulvio.corno, luigi.derussis, andrea.marcelli, teodoro.montanaro}@polito.it

Manuscript received November 17, 2017; revised May 17, 2018 and July 3, 2018.

by the user terminal without computing or battery overheads, to support network operators in managing the distribution of network resources.

The proposed approach is based on a two stages methodology. The former extracts a set of meaningful locations for each user, using non-invasive information collected from the end-user terminal, i.e., temporal information, mobile network cells, and properly anonymized Wi-Fi Service Set IDentifier (SSID). The latter analyzes the data to build a stochastic Markov chain to predict the users movements and their bandwidth demands.

To evaluate the proposed methodology, we tested it with one of the largest community labeled mobile datasets, developed under the "CrowdSignals.io" initiative [10], and we present positive and promising results concerning the prediction capabilities of the model.

The rest of this paper is organized as follows. Section II describes in details the proposed methodology. Section III illustrates the dataset, while Section IV covers experimental results and performance evaluation. Related works about user locations estimation are reported in Section V. Finally, Section VI concludes the paper and illustrates future work.

II. Method

This paper proposes an unsupervised methodology to support network providers in predicting user network demand, relying on a minimal subset of non-invasive data to be collected from end-user terminals. It exploits the temporal correlation between wireless networks visible by the terminal, in particular, all the visible Wi-Fi networks and the connected cellular base station at a given time. The approach has been designed to be applicable to any dataset where such wireless information is available, and it is based on a two stages methodology.

In the first stage, a clustering approach is used to infer a set of *meaningful network locations MNL* for each user and for each time period. We define a meaningful network location as a specialization of the concept of meaningful location proposed in [7], as a recurring network (or a set of networks) that the user is connected to, in a significant period of time. It does not imply a geographical location, but just the position of the user from the point of view of the network operator. When a user is in a given MNL, he will consume a given amount of traffic, and the knowledge of the MNLs of each user, as well as their evolution, allows network resource pre-allocation, reservation and caching.

The resulting clusters represent frequently occurring meaningful network locations. In order to increase the robustness of the model, the clusters are jointly computed by analyzing in parallel Wi-Fi-detected locations and cellular-detected locations, and the maximum accord between the two clustering results is sought. In the second stage, the computed clusters are used to build a probabilistic Markov chain, which is used to estimate the most likely meaningful location where the user could move, depending on the current location and time.

In the following paragraphs both phases will be presented.

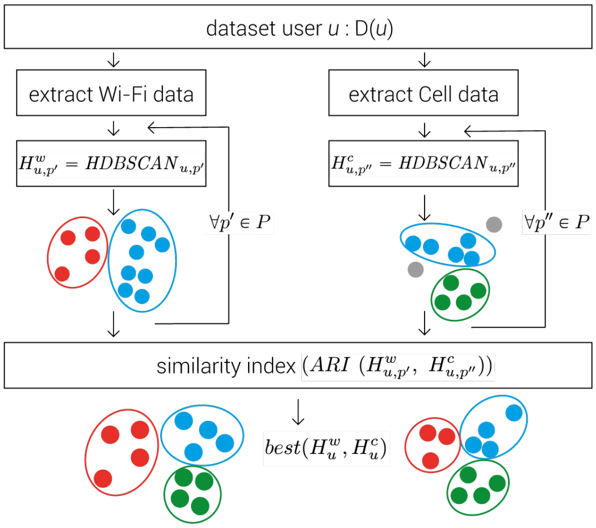
dataset user u : D(u)
extract Wi-Fi data
extract Cell data
$H^w_{u,p'} = HDBSCAN_{u,p'}$
$H^c_{u,p''} = HDBSCAN_{u,p''}$
$\forall p' \in P$
$\forall p'' \in P$
similarity index $(ARI\ (H^w_{u,p'},\ H^c_{u,p''}))$
$best(H^w_u, H^c_u)$

Fig. 1. illustrates the first phase of the proposed methodology: the *conjoined clustering*. A clustering algorithm (HDBSCAN) is iteratively applied with different combination of parameters $p', p'' \in P$ to the Wi-Fi and Cell data of user u separately, then the maximum accord between the two clustering results is sought ($best(H^w_u, H^c_u)$).

A. Infer meaningful user locations

The first phase of the proposed approach consists in extracting meaningful network locations by exploiting a completely unsupervised methodology. Figure 1 illustrates the process.

For each user $u \in U$, Wi-Fi and cellular (Cell) data are treated distinctly, using a clustering algorithm to automatically infer groups of meaningful locations related to the user daily activities. Having two independent sources allows us to increase the robustness of the unsupervised learning, by leveraging self-consistency of the detected clusters. Indeed, an external clustering index is used to compare the structure of the two cluster sets, and the entire clustering process is iterated to maximize the value of such an index.

From the point of view of the network, the impact coming from any user depends on the user location and activity (i.e., bandwidth consumed in that location). Locations, on the other hand, tend to be dependent on temporal variables (hour, day of week, etc.), due to user habits. For these reasons, the clusters are defined in terms of spatial and temporal variables.

From the general data set DS, we extract the entries $D(u)$, consisting of all data samples related to user u:

$$D(u) = \{d \in DS \mid d.user = u\}$$

The set of features $F(u)$ used by the clustering algorithm for user u are a combination of *spatial features* $F_s(d)$ and *temporal features* $F_t(d)$, are extracted from each entry d:

$$F(u) = \{\langle F_s(d),\ F_t(d)\rangle \mid d \in D(u)\}$$

Spatial features $F_s(d)$ are encoded as a high-dimensional Boolean vector in $E_s(F_s(d), u)$, where all the possible networks N visited by user u ($N = \{d.network \mid d \in D(u)\}$) are represented in a one-hot encoding. Given the available network $n \in N$, we set $E_s[n] = 1$ if the device is covered by the network, $E_s[n] = 0$ otherwise.

$$E_s(F_s(d), u) = \bigtimes_{n \in N} n \in F_s(d)$$

Temporal features $F_t(d)$ are encoded in $E_t(F_t(d))$ as a low-dimensional integer vector, where each component represents one kind of periodicity (e.g., daily, weekly). Specific details of the representation of $F_t(d)$ are dependent on the dataset, and the ones adopted in this paper are reported in Section III.

The final encoding $E(u)$ for user u is defined as:

$$E(u) = \{\langle E_s(F_s(d), N),\ E_t(F_t(d))\rangle \mid d \in D(u)\}$$

We adopt a *density-based* clustering approach to locate regions of high density, surrounded by regions of low density. Differently from other clustering methods, density-based algorithms can effectively discover clusters of arbitrary shape and filter out outliers, increasing cluster homogeneity. Additionally, the number of expected clusters to be found in the data is not required in advance, and in many practical cases, such as the number of meaningful user locations, this is hard to be defined *a priori*. In low-dimensional spaces, the time complexity of density-based clustering can be as low as $O(n \log n)$, while its space requirement is $O(n)$, making it applicable to large datasets. In 2013, Campello et al. [11] proposed Hierarchical Density-Based Spatial Clustering of Applications with Noise (HDBSCAN), which practically iterates DBSCAN over several values of the parameter ϵ, which specifies how close points should be to each other to be considered a part of a cluster, and integrates the results to find a clustering that provides the best stability over ϵ. This allows HDBSCAN to find clusters of varying densities, and be more robust to parameter selection.

HDBSCAN relies on two parameters $p = (mss, ms)$: the choice of a *minimum cluster size* (mss), which determines the smallest allowed size for a cluster, i.e., we considered only clusters with a minimum of mss samples as a meaningful user location; and *minimum samples* (ms), which influences the expected density of clusters in the results. As a matter of fact, a higher value of ms restricts clusters to more dense areas, but it also increases the number of outliers.

Encoded features $E(u)$ are firstly normalized as $\widetilde{E}(u)$, to have zero mean and unit variance. Then, the high dimensionality of $\widetilde{E}(u)$, mostly due to the large number of Wi-Fi and cellular networks visited by the user u (e.g., up to thousands), is handled by a linear dimensionality reduction by means of truncated singular value decomposition (SVD), down to a more manageable number of components. The application of SVD is also beneficial for discovering linearly independent combinations of networks and for filtering out rogue or sporadic networks, that are frequent especially in the case of Wi-Fi.

We finally apply HDBSCAN on the normalized and SVD-reduced dataset of user u, with parameters p, that generates a clustering result $H_{u,p}$:

$$H_{u,p} = HDBSCAN\left(SVD\left(\widetilde{E}(u)\right), p\right),\ p \in P$$

$$P = \{mss_1, \ldots, mss_m\} \times \{ms_1, \ldots, ms_q\}$$

The clustering result is represented by the function $H_{u,p}(d)$, that associates each entry d of the dataset of user u ($D(u)$) to a cluster c_i.

$$H_{u,p} : D(u) \to \{c_i \mid \forall i\}, H_{u,p}(d) = c_i$$

In order to optimize the parameter selection p, the clustering algorithm is iteratively applied through a set of possible values P, trying to find the pair of assignments that best fits both the Wi-Fi (H_u^w) and the cellular data (H_u^c) at the same time. We call this approach a *conjoined clustering*, since it aims at evolving in parallel two sets of clusters with high correlation among them. In our experiments we iterate through values of $mss \in \{50, 100\}$ and $ms \in \{10, 20, 30, 40, 50\}$.

For each combination of p' and p'' (100 combinations in total), we measure the similarity of the two assignments H_u^w and H_u^c, and we save the best resulting clusters $best(H_u^w, H_u^c)$ that maximize such similarity.

$$best(H_u^w, H_u^c) = \underset{p' \in P,\ p'' \in P}{argmax}\left(ARI\ (H_{u,p'}^w,\ H_{u,p''}^c)\right)$$

Cluster similarity is measured by using the *Adjusted Rand Index* (*ARI*) [12]. ARI is defined as the number of pairs of items that are either both in the same cluster or both in different clusters in the two partitions, normalized over the total number of pairs of items. The index lies between 0 and 1: when two partitions perfectly agree, the ARI achieves the maximum value 1, and more in general a larger ARI means a higher agreement between two partitions. Moreover, ARI allows measuring the level of agreement even when the compared partitions have different numbers of clusters. The resulting best matching clusters are therefore assumed to approximate the meaningful network locations for user u.

At this point, for each best-matching cluster c_i, it is possible to compute the corresponding resource demand R_{u,c_i}:

$$R_{u,c_i} = \sum_{d \in D(u) \wedge H_u(d) = c_i} d.traffic$$

At the end of this phase, we have the following available results:

- Two sets of clusters $\{c_i^w \mid \forall i\}$ and $\{c_j^c \mid \forall j\}$, extracted from two independent sensor data streams (i.e., Wi-Fi and cellular data), and cross-validated thanks to the maximization of the ARI index, that approximate the meaningful network locations for user u. Each cluster contains a spatial and a temporal component $(F_s(d),\ F_t(d))$.
- The resource demand R_{u,c_i} (e.g., bandwidth) consumed by a user u in the given cluster (or MNL) c_i, obtained by aggregating resource demands for all the original data entries assigned to that cluster.

B. Build a stochastic Markov chain

The clustering information is useful to have a map of the actual location of the user, and its associated resource consumption. From the network operator point of view, the most useful information would be to *predict* the future cluster, in order to have, in the short term, the expected user location and the associated resource demands.

For this reason, we build a stochastic Markov chain, with a time-step of one hour, that encodes the transition probabilities among different clusters. We adopt a "time-homogenous" Markov Chain [13], where we assume that transition probabilities are not time-dependent. This assumption is intuitively justified by the structure of each Markov state (that contains both a spatial and a temporal component) and by the clustering process: whenever the behavior of a user depends on a temporal feature (e.g., the hour of the day), we will have two separate states.

Again, two models are built, one for Wi-Fi data, and the other for cellular data.

As a pre-processing step, time-related information in clusters must be extracted. The items in the best-matching clusters are grouped by *hour* (h) and *day of the month* (dm), and the most frequent cluster in each temporal interval is extracted and saved. Specifically, we defined two *frequent cluster* functions $pos^w(u,h,dm)$ and $pos^c(u,h,dm)$, that return, for each time interval, the most representative cluster for the wireless and cell network, respectively.

$$pos^w(u,h,dm) = \\ \underset{c_i^w \in H_u^w}{argmax} \, |\{H_u^w(d) = c_i : d.h = h \wedge d.dm = dm\}|$$

$$pos^c(u,h,dm) = \\ \underset{c_i^c \in H_u^c}{argmax} \, |\{H_u^c(d) = c_i : d.h = h \wedge d.dm = dm\}|$$

The two time-mapping functions $pos^w(u,h,dm)$ and $pos^c(u,h,dm)$ therefore predict the most frequent cluster (MNL), depending on time variables.

The Markov chain for the Wi-Fi data is defined as follows (the one related to the cellular data is similarly constructed):

- each distinct value of $pos^w(u,h,dm)$ is a state $s_u^w \in S_u^w$: it represents the user being in a given *meaningful network location* in a given time period (identified by (h,dm)):
$$S_u^w = \{pos^w(u,h,dm) \mid \forall h, dm\}$$
- transitions between states are created for every adjacent time step (hour → hour + 1 in the same day)
$$\{\langle pos^w(u,h,dm), pos^w(u,h+1,dm)\rangle \mid \forall h, dm\}$$
- transition probabilities $p_u^w(s_1,s_2)$ are assigned according the frequency of the transition between the clusters observed in the temporal stream of the dataset. In particular:

$$p_u^w(s_1,s_2) \propto |\{pos^w(u,h,dm) = s_1 \\ \wedge \; pos^w(u,h+1,dm) = s_2 \mid \forall h, dm\}|$$

The Markov chain, therefore, models the stochastic evolution of the user across the set of meaningful locations associated to his habits. For each state we also know the associated resource demand R_{u,c_i}, thus stochastically modeling the evolution of network resource demands, too. In general user habits evolve over time, hence the model is useful to predict user demands over a limited period of time; after that, it should be updated with new incoming data [8].

III. Data collection

A. Dataset

The clustering approach and the Markov chain described in the previous section allow the definition of a user mobility model able to support network providers in predicting user network demand and, consequently, to improve the resource distribution strategies. To validate the proposed methodology, we used one of the largest community labeled mobile datasets, developed under the "CrowdSignals.io" initiative [10].

The "CrowdSignals.io" dataset contains longitudinal mobile and sensor data recorded from smart-phones and smartwatches available to the community[1]. It was selected as the most recent available dataset that contains almost all the data that can be acquired through mobile devices.

As of May 2018, the "CrowdSignals.io" initiative is still in a "pilot" phase, consequently, the dataset contains data collected from 40 users among 30 days of actual usage, only. Moreover, by considering the high variability of the involved devices and the possibility of disabling some device features (e.g., on some Android devices, the operating system may deactivate some features depending on the current battery level), some information is not available for all the 40 users. For this reason, after a pre-processing phase on the dataset, the present work uses data from 11 users. That portion of the dataset, in fact, is the only one that contains all the needed information.

As already explained in the previous sections, only the Wi-Fi and the cellular network data fields were selected among all the available information, as they constitute the least invasive user information that network providers can easily access and that can be captured without computing and/or battery overhead.

Wi-Fi data consists of the full list of wireless networks (identified by their SSID, which was anonymized in the dataset) visible by the user's smartphone at a given time, independently from its actual connected network and from the signal strengths of those networks. Therefore, over the lifetime of the data collection, each user may have seen thousands of different SSIDs. Table I reports the number of available wireless network samples, the total number of distinct SSIDs for each user.

TABLE I
Wi-Fi data for the selected subset of users

User Id	N. of Samples	N. of SSID	N. of Hours
1	9,524	2,614	163
2	23,822	7,654	266
10	15,164	3,145	243
21	41,677	3,322	414
28	34,633	6,572	332
29	4,972	9,682	223
30	2,920	3,364	96
31	13,955	3,003	248
37	15,990	3,224	168
39	13,372	7,082	346
41	19,075	2,848	253

Cellular data samples the Base Station Identifier (BID) of the cellular towers to which the user smart-phone was

[1] The dataset can be obtained from http://crowdsignals.io/

connected. At most one BID is available per each sample (i.e., base stations that may be in range but are not connected are not saved), therefore the total number of distinct BIDs is in the order of dozens, depending on the movements of each user. Table II reports the details about cellular data.

TABLE II
CELLULAR DATA FOR THE SELECTED SUBSET OF USERS

User Id	N. of samples	N. of BID	N. of hours
1	6,616	58	164
2	34,870	10	279
10	50,005	18	321
21	4,886	55	155
28	28,639	9	331
29	85,051	17	336
30	32,195	18	168
31	33,978	12	305
37	17,211	15	167
39	31,733	17	320
41	27,768	46	241

Finally, Tables I and II report the number of hours in which samples are available. It is possible to notice that the overlap between Wi-Fi data and cellular data is not complete, as it depends on the actually collected data.

B. Feature selection

The accurate selection of features is a crucial step in every machine learning approach. In this section, we describe how the general method presented in Section II is applied to the just presented dataset, taking into account its limitations and characteristics. From the available data, we extract the features F, composed of a spatial component F_s and a temporal component F_t.

Spatial Features F_s: Spatial features are encoded differently in the Wi-Fi and cellular cases. For the Wi-Fi case, the feature vector F_s^w, is encoded in E_s^w using the *one-hot encoding* of the set of SSIDs; this vector may grow to thousands of components, and it may contain one or more "1" values in each sample. For the cellular case, instead, F_s^c is encoded in E_s^c using the *one-hot encoding* of the BID; such a vector has exactly one "1" per sample, corresponding to the uniquely connected base station.

Temporal Features F_t: Temporal features represent the time instant where the samples were taken, and are decomposed to discern possible periodicity effects in the data. In particular, F_t contains seven time-related information fields, automatically derived from the time-stamp of each sample: *month*, *day* (of the month), *day_of_the_week*, *hour*, *minute*, *second* and, finally, *time_of_the_day* that breaks the day according to the assignments reported in Table III.

TABLE III
TIME_OF_THE_DAY FEATURE VALUES ASSIGNMENT

Day hours	Day activity
00-05	sleep
06-08	breakfast
09-11	morning activities
12-13	lunch
14-17	afternoon activities
18-19	evening activities
20-21	dinner
22-23	night activities

As explained in Section II, encoded features E^w and E^c are then normalized, to have zero mean and unit variance, and SVD-reduced to lower the number of components. The choice of the maximum number of components is related to the dataset size and the cardinality of distinct wireless networks visited by each user. In our experiments we used a maximum of 100 components, as it delivered best results. Similarly, the choice of the distance to use during cluster analysis is tied to the type and the dimension of selected features, and we experimentally found that the Euclidean distance delivered the best performance.

IV. EXPERIMENTAL RESULTS

We implemented the proposed methodology in a few thousands lines of Python 3.6 code[2]. All tests were performed on a server equipped with a 4-core Intel i5 processor (i5-2500 CPU @3.30 GHz), 16 GB of RAM, and running Ubuntu 16.04.3 LTS. For the clustering algorithm, we exploited a high performance implementation of HDBSCAN from L. McInnes [14] and the Scikit-learn library [15] for the data analysis and validation process.

A. Conjoined clustering

The conjoined clustering algorithm described in Section II-A was run separately for each user u to generate a pair of cluster assignments H_u^w and H_u^c. The HDBSCAN algorithm is memory-friendly: in our tests the memory usage never exceeded 1 GB, and the run time of each HDBSCAN run was always less than a minute.

The results of the HDBSCAN algorithm were then filtered, thanks to the frequent cluster functions $pos^w(u, h, dm)$ and $pos^c(u, h, dm)$, by selecting only the most frequent clusters for each one-hour time step.

For example, Table IV shows some of the clusters assigned to User 1. For every Day/Hour combination, the most frequent Wi-Fi cluster and the most frequent Cellular cluster are extracted. In some cases, the data points might not belong to any cluster, and in these cases the cluster is marked as "-1".

Table V shows for each selected user the number of frequent clusters inferred from Wi-Fi and Cellular data. The rightmost column highlights the number of hours of data acquisition that the two subsets of data have in common. The size of the cluster is larger for Wi-Fi in some cases and for Cellular in other cases, but this depends on the actual user data.

The dataset also provides the amount of network resources consumed by the users in each data sample. After the conjoined clustering, we may estimate the network demand R_{u,c_i} for a user u in a specific cluster c_i (i.e., in a specific meaningful location). For example, Table VI shows the traffic demands for

[2] The source code used to run the experiments is available at https://github.com/jimmy-sonny/CrowdSignals.io. The code is licensed under the 2-Clause BSD license.

TABLE IV
EXAMPLE OF CLUSTER ASSIGNMENT FROM USER 1

Day	Hour	WI-Fi Cluster	Cellular Cluster
..	..	..	..
23	20	11	17
23	21	11	17
23	22	11	17
23	23	11	17
..	..	..	..
24	19	12	10
24	20	12	10
24	21	12	10
24	22	-1	-1
24	23	-1	-1
..	..	..	..

TABLE V
CLUSTERING RESULTS FOR THE SELECTED SUBSET OF USERS

User Id	N. Wi-Fi Clusters	N. Cell Clusters	N. hours in common
1	12	12	160
2	22	23	166
10	30	29	158
21	32	10	79
28	10	28	168
29	8	8	145
30	8	7	97
31	14	31	156
37	20	29	167
39	21	18	168
41	9	11	168

User 1 in his various meaningful network locations, using Wi-Fi data, both in average terms (i.e., by taking into account the time spent in the locations) and in total. The last line reports the network traffic attributed to data samples that were not included in any cluster (*outliers*): 73.86% of the total traffic was accounted by clusters, and therefore may be predicted by the model.

TABLE VI
NETWORK RESOURCE DEMAND FOR USER 1 WI-FI CLUSTERS

Cluster #	AVG Network Traffic (MB)	Total Network traffic
0	8.02	16.03
1	30.05	30.05
2	0.00	0.00
3	17.55	210.65
4	5.44	16.32
5	0.00	0.00
6	0.00	0.00
7	1.50	7.53
8	0.00	0.00
9	25.54	1302.82
10	14.16	42.47
11	0.035	0.31
outliers	10.66	575.54

The same data is reported graphically in Figure 2 for User 1, and in Figure 3 for User 39, where the clusters with high network usage have been highlighted in red. This shows how a network operator may optimize its resources, according to the overall demand, and in a completely automatic way.

Analyzing the result of clustering is challenging as it involves the use of subjective criteria of optimality, specific to a particular application. Therefore, no commonly accepted standard for validating the output of a clustering procedure exists [16].

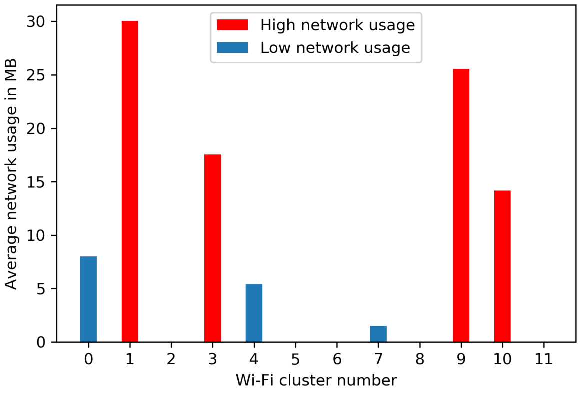

Fig. 2. Network resource demand for User 1 Wi-Fi clusters

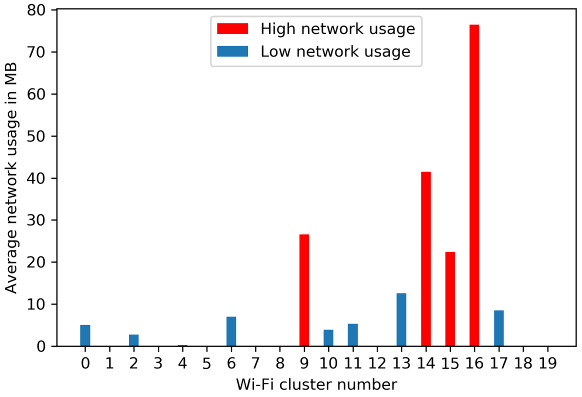

Fig. 3. Network resource demand for User 39 Wi-Fi clusters

As mentioned in Section II-A, we adopt the Adjusted Rand Index (ARI) to estimate cluster similarity, and therefore ensure the self-consistency of the results of the conjoined clustering. In addition to ARI, we also computed the following indexes of the quality of the clustering result, as proposed by Rosemberg and Hirschberg [17]:

- *Homogeneity*, which measures whether its clusters contain only data points which are members of a single class;
- *Completeness*, which measures whether all the data points that are members of a given class are elements of the same cluster;
- *V-measure*, measured as the weighted harmonic mean of homogeneity and completeness; this is useful since homogeneity and completeness of a clustering solution run roughly in opposition: increasing the homogeneity of a clustering solution often results in decreasing its completeness.

Table VII reports the evaluation of clustering assignment from the external clustering indexes ARI, Homogeneity, Com-

TABLE VII
UNSUPERVISED EXTERNAL INDEXES COMPARISON

User Id	ARI	Homogeneity	Completeness	V-Index
1	0.35	0.73	0.68	0.70
2	0.12	0.76	0.72	0.74
10	0.20	0.27	0.27	0.27
21	0.06	0.16	0.23	0.19
28	0.08	0.13	0.06	0.08
29	0.02	0.08	0.07	0.07
30	0.01	0.18	0.20	0.19
31	0.10	0.24	0.25	0.25
37	0.14	0.45	0.23	0.30
39	0.23	0.84	0.77	0.80
41	0.08	0.45	0.50	0.48

pleteness, and V-measure.

The computed values of the ARI were satisfying, especially for a dataset collected in-the-wild and with various shortcomings. Only users 21, 28, 29, and 30 exhibited too low values. In most cases, even a low ARI value is compensated by a significantly better value of Homogeneity.

B. Markov chain

While the clustering results allow the representation, in a synthetic way, of the behavior of the user, the Markov chain, computed as described in Section II-B, allows the prediction of her near-future behavior (in the next hour), in terms of location and of network demand.

The computation of the Markov chain relies on the frequent clusters assigned to each time period through the time-mapping functions $pos^w(u, h, dm)$ and $pos^c(u, h, dm)$, and each cluster encodes a single state in the Markov chain. The Markov model includes transitions every hour, with the probability of getting to a specific cluster, given the previous one.

As an example, Figures 4 and 5 illustrate a heatmap representing the transition probability for the Markov chains computed for User 1, using the Wi-Fi data and the Cellular data, respectively.

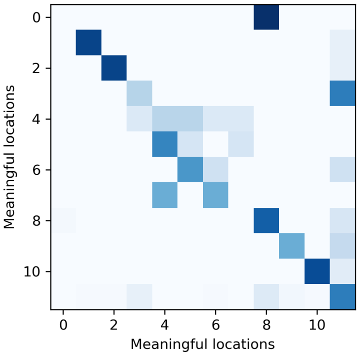

Fig. 4. Markov model from the Wi-Fi data of User 1

The availability of such models allows the network operator to predict the future meaningful locations of the user, given his current location (measured in real time). The operator might rely on the most-likely location, only, or might consider all the likely future locations, to pre-allocate network resources to the closest EDC. Since each cluster is associated with a specific network demand, resources can be allocated with a fine level of granularity.

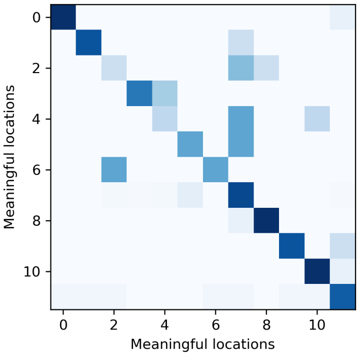

Fig. 5. Markov model from the Cellular data of User 1

Moreover, the operator can benefit from two separate prediction models (Wi-Fi and Cellular), and select the most suitable one according to the available information.

We can also observe that, thanks to the clustering process, the Markov chain associated to each user is really small: the operator may therefore maintain prediction models for each user, within acceptable computational resources.

V. Related works

The prediction of network resource demand has been subject of extensive studies in the literature, but only a few works exploit the estimation of future user movements. Tan *et al.* [4], for example, propose a novel location-aware load prediction approach which deals with user mobility by correlating load fluctuations of EDCs in physical proximity. They exploited GPS coordinates of 536 taxis collected over 25 days to generate the historical load time series for each EDC and declared that their method outperforms state-of-the-art location-unaware prediction methods by up to 4.3%.

On the other hand, the estimation of user future movements through two-stage approaches has been subject of existing studies in the literature, nonetheless, they usually exploit *invasive* user information captured with relevant computing and/or battery overhead, and require the installation of specific applications on the user terminals.

Pant *et al.* [18] present a two-stage method that uses a) a Varied-K Means clustering technique to establish the user meaningful locations from GPS data, and b) Hidden Markov Model techniques to predict user's future movements based on the user's past historical data, i.e., weekday and time period within the day. A similar work is performed by Ashbrook *et al.* [19], [20]. They present a system that automatically clusters GPS data taken over an extended period of time into meaningful locations. These locations are then incorporated into a Markov model that can be used to predict user locations. Their prediction is dependent on the user's past location, but, differently from the previous work, it does not use day and time. Another similar proposal is presented by Yang *et al.* [21]. It, also, presents a two-stage approach for predicting user locations based on GPS data. However, they employ DBSCAN techniques, instead of K-means, to cluster GPS data.

As can be depicted from the presented descriptions, all the presented works are similar: they use GPS information to cluster user meaningful locations as a the first step and, then, they provide a model representative of user movements. However, the GPS information used by all of them is both a) battery-draining and b) cannot be acquired by network operators without specific license agreements (and ad-hoc installation of applications on the user terminals). Therefore, the following paragraphs analyze existing solutions to cluster user meaningful locations without using GPS information and/or other battery-draining data.

From the literature, three of the most used information in clustering user meaningful locations without causing computing and/or battery overhead are: "precision sensors" data, anonymized Wi-Fi SSID, and cellular network data. The "precision sensors" data includes information acquirable through accelerometer, gyroscope, digital compass, microphone, and/or magnetometer sensors. They are used in various existing works (e.g., [22]–[24]), but, even though some techniques were proposed to reduce battery consumption in their acquisition [22], they are not, in any case, acquirable by network operators without additional license agreements. Thus, we will concentrate on the other two, i.e., anonymized Wi-Fi SSID and the cellular network data, that are not battery-draining data and that are easily accessible by network operators.

Regarding the anonymized Wi-Fi SSID, Nguyen *et al.* [25] present a method to determine significant locations by clustering Wi-Fi access points in close proximity using the Affinity Propagation algorithm, an unsupervised algorithm that is able to cluster the samples without knowing the cluster number in advance. They demonstrate the reliability of their approach on the Wi-Fi data obtained from the Mobile Data Challenge (MDC) dataset: the correlation between the inferred locations with the user's visited locations included in the dataset confirms the validity of their approach. Similarly, Zhao et al. [26] use Wi-Fi scanlists that are clustered into a set of stay places by a clustering method. They, also, confirm that techniques that use anonymized Wi-Fi SSID to cluster user meaningful locations are a promising approach, to be further investigated.

For what concern the cellular network data, instead, Fanourakis et al. [27] present a lightweight method to form semantically meaningful clusters of cellular IDs from cellular ID sequences. The method was preliminarily tested on 15 weeks of data, collected from one real user in her natural daily environments. Furthermore, Isaacman et al. [28] propose new techniques based on clustering for analyzing anonymized cellular network data to identify generally important location. Results presented in both works are promising as a step towards a lightweight method to cluster cellular IDs into meaningful information for the user neighborhoods.

The problem of dynamically adjust EDC capacity and optimize resource utilization is similar to the dynamic resource allocation in the fog computing architecture. Zhang *et al.* [6] propose an optimization framework, based on the "Stackelberg games" and the "many-to-many matching" algorithm, to achieve an optimal resource allocation schema in a fog architecture. On the other hand, Ni *et al.* [5] present a dynamic resource allocation strategy for fog computing, based on Petri nets (PTPNs), to improve the efficiency of fog resources utilization and to satisfy user QoS. Since a fog application can be decomposed into several tasks to be executed on fog resources, they propose a performance prediction model to reduce the task response time, maximize the resource utilization, and lower the global costs.

VI. Conclusion and future work

Future generation networks will face the challenge of increased user mobility, and pressing resource demands due to IoT services, to be served by a dynamic reconfigurable network. This scenario requires intelligent network mechanisms able to adapt to the evolving user demands, even in real time, and shift resource allocation according to user behavior.

This work proposes a predictive model, which is based on non-intrusive monitoring of user data connections and consumed network resources, able to identify the *meaningful network locations* of each user, and its associated resource demand. The model allows a network operator to estimate and pre-allocate network resources with a one-hour time step, according to the predicted demand customized to each user's behavior. Quantitative results over the "CrowdSignals.io" dataset validate the proposed model and show its effectiveness.

In future work, we aim at exploiting a Markov model that integrates Wi-Fi and cellular meaningful network locations, thanks to the correspondence between the clusters given by the conjoined clustering. Moreover we plan to extensively test the proposed methodology on other available datasets.

We will also investigate alternative unsupervised methodologies, in particular we will try to model meaningful network locations using embeddings. In a similar way, we will explore alternative approaches to predict the future users' network demands, testing the effectiveness of the adoption of machine learning classification algorithms in this field too.

Acknowledgment

The Ph.D. programs of A. Marcelli and T. Montanaro are supported by a fellowship from TIM (Telecom Italia).

References

[1] A. Brogi and S. Forti, "Qos-aware deployment of iot applications through the fog," *IEEE Internet of Things Journal*, vol. 4, no. 5, pp. 1185–1192, Oct 2017.

[2] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, "Fog computing and its role in the internet of things," in *Proceedings of the First Edition of the MCC Workshop on Mobile Cloud Computing*, ser. MCC '12. New York, NY, USA: ACM, 2012, pp. 13–16.

[3] B. Liang, *Mobile edge computing*. Cambridge University Press, 2017.

[4] C. N. L. Tan, C. Klein, and E. Elmroth, "Location-aware load prediction in edge data centers," in *2017 Second International Conference on Fog and Mobile Edge Computing (FMEC)*, May 2017, pp. 25–31.

[5] L. Ni, J. Zhang, C. Jiang, C. Yan, and K. Yu, "Resource allocation strategy in fog computing based on priced timed petri nets," *IEEE Internet of Things Journal*, vol. 4, no. 5, pp. 1216–1228, 2017.

[6] H. Zhang, Y. Xiao, S. Bu, D. Niyato, F. R. Yu, and Z. Han, “Computing resource allocation in three-tier iot fog networks: A joint optimization approach combining stackelberg game and matching,” *IEEE Internet of Things Journal*, vol. 4, no. 5, pp. 1204–1215, 2017.

[7] P. Nurmi and J. Koolwaaij, “Identifying meaningful locations,” in *2006 3rd Annual International Conference on Mobile and Ubiquitous Systems - Workshops*, July 2006, pp. 1–8.

[8] X. Cheng, L. Fang, L. Yang, and S. Cui, “Mobile big data: the fuel for data-driven wireless,” *IEEE Internet of Things Journal*, vol. 4, no. 5, pp. 1489–1516, 2017.

[9] X. Cheng, L. Fang, X. Hong, and L. Yang, “Exploiting mobile big data: Sources, features, and applications,” *IEEE Network*, vol. 31, no. 1, pp. 72–79, 2017.

[10] E. Welbourne and E. M. Tapia, “Crowdsignals: A call to crowdfund the community’s largest mobile dataset,” in *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication*, ser. UbiComp ’14 Adjunct. New York, NY, USA: ACM, 2014, pp. 873–877.

[11] R. J. Campello, D. Moulavi, and J. Sander, “Density-based clustering based on hierarchical density estimates,” in *Pacific-Asia Conference on Knowledge Discovery and Data Mining*. Springer, 2013, pp. 160–172.

[12] L. Hubert and P. Arabie, “Comparing partitions,” *Journal of Classification*, vol. 2, no. 1, pp. 193–218, dec 1985. [Online]. Available: https://doi.org/10.1007/bf01908075

[13] S. Karlin, *A First Course in Stochastic Processes*. Academic Press, 2014.

[14] L. McInnes, J. Healy, and S. Astels, “hdbscan: Hierarchical density based clustering,” *The Journal of Open Source Software*, vol. 2, no. 11, mar 2017. [Online]. Available: https://doi.org/10.21105%2Fjoss.00205

[15] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, “Scikit-learn: Machine learning in Python,” *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[16] A. K. Jain, M. N. Murty, and P. J. Flynn, “Data clustering: a review,” *ACM computing surveys (CSUR)*, vol. 31, no. 3, pp. 264–323, 1999.

[17] A. Rosenberg and J. Hirschberg, “V-measure: A conditional entropy-based external cluster evaluation measure.” in *Proceedings of EMNLP*, vol. 7, 2007, pp. 410–420.

[18] N. Pant and R. Elmasri, “Detecting meaningful places and predicting locations using varied k-means and hidden markov model,” in *17th SIAM International Conference on Data Mining (SDM 2017), 3rd International Workshop on Machine Learning Methods for Recommender Systems, At Houston, Texas, USA*, April 2017.

[19] D. Ashbrook and T. Starner, “Learning significant locations and predicting user movement with gps,” in *Proceedings. Sixth International Symposium on Wearable Computers,*, 2002, pp. 101–108.

[20] ——, “Using gps to learn significant locations and predict movement across multiple users,” *Personal Ubiquitous Comput.*, vol. 7, no. 5, pp. 275–286, Oct. 2003. [Online]. Available: http://dx.doi.org/10.1007/s00779-003-0240-0

[21] J. Yang, J. Xu, M. Xu, N. Zheng, and Y. Chen, “Predicting next location using a variable order markov model,” in *Proceedings of the 5th ACM SIGSPATIAL International Workshop on GeoStreaming*, ser. IWGS ’14. New York, NY, USA: ACM, 2014, pp. 37–42. [Online]. Available: http://doi.acm.org/10.1145/2676552.2676557

[22] Y. Han, J.-M. Kang, S. seok Seo, A. Mehaoua, and J. W. K. Hong, “An energy efficient user context collection method for smartphones,” in *2013 15th Asia-Pacific Network Operations and Management Symposium (APNOMS)*, Sept 2013, pp. 1–6.

[23] Y.-S. Lee and S.-B. Cho, “Activity recognition using hierarchical hidden markov models on a smartphone with 3d accelerometer,” in *Proceedings of the 6th International Conference on Hybrid Artificial Intelligent Systems - Volume Part I*, ser. HAIS’11. Springer-Verlag, 2011, pp. 460–467.

[24] L. Garbe, “System identifies user location without gps or wi-fi,” *Computer*, vol. 44, no. 11, pp. 15–17, 2011.

[25] T.-B. Nguyen, T. Nguyen, W. Luo, S. Venkatesh, and D. Phung, “Unsupervised inference of significant locations from wifi data for understanding human dynamics,” in *Proceedings of the 13th International Conference on Mobile and Ubiquitous Multimedia*, ser. MUM ’14. New York, NY, USA: ACM, 2014, pp. 232–235.

[26] S. Zhao, Z. Zhao, Y. Zhao, R. Huang, S. Li, and G. Pan, “Discovering people’s life patterns from anonymized wifi scanlists,” in *2014 IEEE 11th Intl Conf on Ubiquitous Intelligence and Computing and 2014 IEEE 11th Intl Conf on Autonomic and Trusted Computing and 2014 IEEE 14th Intl Conf on Scalable Computing and Communications and Its Associated Workshops*, Dec 2014, pp. 276–283.

[27] F. Marios and K. Wac, “Lightweight clustering of cell ids into meaningful neighbourhoods,” in *Proceedings of the Seventh International IFIP Working Conference on Performance and Security Modelling and Evaluation of Cooperative Heterogeneous Networks (HET-NETs)*, 2013.

[28] S. Isaacman, R. Becker, R. Cáceres, S. Kobourov, M. Martonosi, J. Rowland, and A. Varshavsky, “Identifying important places in people’s lives from cellular network data,” in *Proceedings of the 9th International Conference on Pervasive Computing*, ser. Pervasive’11. Berlin, Heidelberg: Springer-Verlag, 2011, pp. 133–151.

Fulvio Corno is an Associate Professor at the Department of Control and Computer Engineering of Politecnico di Torino, Italy, since 2002. He is the leader of the e-Lite research group, where he focuses on Ambient Intelligence systems by integrating novel interaction modalities with IoT architectures. He is a member of the IEEE, of the IEEE Computer Society and of the ACM. Contact him at fulvio.corno@polito.it

Luigi De Russis is an Assistant Professor in the Department of Control and Computer Engineering at Politecnico di Torino, Italy, since 2018. His research interests include Human-Computer Interaction (HCI) and Ambient Intelligence. He received a Ph.D. in Computer and Control Engineering from Politecnico di Torino in 2014. He is a member of IEEE, the IEEE Computer Society, and ACM. Contact him at luigi.derussis@polito.it

Andrea Marcelli received his M.Sc. degree in Computer Engineering from Politecnico of Torino, Italy, in 2015. Currently he is a Ph.D. student in Computer and Control Engineering at the same institute and member of the CAD group. His research interests include malware analysis, semi-supervised modeling, machine learning and optimization problems, with main applications in computer security. He is an IEEE Student Member. Contact him at andrea.marcelli@polito.it

Teodoro Montanaro is a Ph.D. student in Computer Engineering of Politecnico di Torino, Italy since 2014. His current research focuses on the investigation of the intelligence component in Internet of Things (IoT) architectures and applications. He is also an IEEE Student Member. Contact him at teodoro.montanaro@polito.it