Classification of Users' Transportation Modalities in Real Conditions

C. Badii, A.Difino, P. Nesi, I. Paoli, M. Paolucci

University of Florence, Department of Information Engineering, Distributed Systems and Internet Tech lab

DISIT Lab, https://www.disit.org , http://www.sii-mobility.org , https://www.km4city.org <name.surname>@unifi.it

Corresponding Author: Paolo Nesi, paolo.nesi@unifi.it

Abstract — The modern mobile devices and the complete digitalization of the public and private transport networks have allowed to access useful information to understand the user's mean of transportation. This enables a plethora of old and new applications in the fields of sustainable mobility, smart transportation, assistance, and e-health. The precise understanding of the travel means is at the basis of the development of a large range of applications. In this paper, a number of metrics has been identified to understand whether an individual on the move is stationary, walking, on a motorized private or public transport, with the aim of delivering to city users personalized assistance messages for: sustainable mobility, health, and/or for a better and enjoyable life, etc. Differently from the state-of-the-art solutions, the proposed approach has been designed to provide results, and thus collect metrics, in *real operating conditions* (imposed on the mobile devices as: a range of different devices kinds, operating system constraints managing Applications, active battery consumption manager, etc.). The paper reports the whole experimentations and results. The solution has been developed in the context of Sii-Mobility Km4City Research Project infrastructure and tools, GDPR compliant. The same solution has been used in Snap4City mobile Apps with experiments performed in Antwerp and Helsinki.

Keywords—user behavior analysis, smart city, mobile phones, transportation modes, classification model, machine learning.

I. Introduction

With the complete digitalization of the public and private transportation networks, the capability of understanding the users' behavior and the mean of transportation have become important. The presence of GPS, accelerometers, sensors on mobile phones has made possible to create solutions exploiting the users' behavior and context. City users are from at the same time information providers and recipients of personalized suggestions and information [Lv et al., 2018]. The understanding of user behavior is the first step for providing suggestions and assistance to people on the move via mobile phones in smart city putting city users in the loop. For example, to allow the city user to receive suggestion to take more virtuous behavior, consume less energy, making more sustainable their transportation, having a healthier life walking more, saving money parking closer (and to monitor their reaction and acceptance level). The research addressed in this article aims to understand the users' mean of traveling taking into account contextual data and data coming from the phones. The correct classification of transportation means can be also used for providing suggestions in the context of public or private transportation. Thus, the above described problem is reconducted to the classification problem of the transportation modality/mean (car, bus, walk, bike, etc.), exploiting real time data coming from the devices and contextual information. Please note that, the contextual data are strongly different in different part of the city, and also change over time, for example busses have different timeline and paths: so that users are moving in the real space.

As described in the following section of related works, the problem of understanding the mean of traveling of users has been many times addressed, but not working in real operating conditions. Most of them, assume data collected from the mobile phones with high rates and high precision, identifying models only taking data in strongly controlled conditions: such as limited number of device type, limited number of users and directly engaged to keep the mobile app running in foreground, etc.

A. Related works

The problem of classifying users' mean of travelling has been addressed by a number of approaches in different research areas [Prelipcean et al., 2017]: *Location Based Services* (LBS), *Transportation Services* (TSc) and *Human Geography* (HG). The LBS solutions aim to understand the transportation meaning in real-time to provide useful information to the user whenever he/she asks. In TSc approaches, the correct segmentation of a trajectory is privileged with respect to velocity of response: [Biljecki et al., 2013]. The HG approaches focus on the segmentation of a trajectory into parts with domain-specific semantics: it is common to first split trajectories into segments where the object is stationary or moving. In LBS, the transportation means' classification is regarded as an online process: an algorithm that provides the current transportation mode of the user in real-time or quasi real-time. To this end, different types of data/sensors have been exploited: GPS, accelerometer and their combination. [Stenneth et al., 2011] compared five different models using data collected from GPS classifying the users' travelling means in six categories (walk, train, driving, stationary, bus, bike). [Hemminki et al., 2013] proposed a study that involves only accelerometer data. They have obtained an 80.1% accuracy and an 82.1% recall for seven transportation modes, by using both AdaBoost and Decision Tree (two-stages classification). [Yu et al., 2014] have compared three different classifiers (Decision Tree obtaining an 84.81% average accuracy, AdaBoost with a 87.16% average accuracy, and SVMs with a 90.66% average accuracy). [Wang et al., 2010] have considered a small data-set of 12 hours (5544 samples of six transportation modes) from 7 different users, obtaining a 70% accuracy with a Decision Tree algorithm. [Reddy et al., 2010] demonstrated that, taking into account of both GPS and accelerometer the accuracy can be improved. They have achieved a 93.6% precision using a combination of Decision Tree and HiddenMarkov Model (two-stages classification), with both accelerometer and GPS features involved, using a sampling rate made a distinction among different type of non-motorized motion (walking, running, biking), vehicular and random movements, using the accelerometer sensors of mobile. [Manzoni et al., 2010] trained a Decision Tree classification model obtaining an 82.14% accuracy (with a gps and accelerometer sampling frequency rate of 1s and 0.04s respectively). [Ashqar et al., 2018], proposed a two-layer hierarchical classifier to predict five classes of transportation mode (car, bus, walk, run, bike), achieving a 97% accuracy. [Yanyun et al., 2017] presented a Convolutional Neural Networks (CNN) based method to automatically extracting features for the identification of transportation means, thus achieving a 98% accuracy to distinguish between train, bus, car, metro.

In **Table 1**, a summary of the state-of-the-art solutions for understanding the travel means is reported (the table report also additional experiments/papers with respect to those commented above). Almost all the state-of-the-art solutions adopted very **high rates for GPS data acquisition, with limited number of devices**. So that, those solution are almost unfeasible in *real operating conditions*. Mobile operating systems allow to keep the high rates (in the order of seconds) only when applications are running in foreground. In most cases, the precisions provided has been obtained with limited set of devices in unrealistic conditions.

Table 1. Related Work implementation overview.

Authors	**Classes**	**data exploited**	**Sampling**	**#users**	**#features**	**# of device types**	**Precision accuracy**
Wang et al.,2010	Stationary, Walk, Bike, Bus, Car, Metro	Accel	0.03s (accel)	7	23	**1**	70
Manzoni et al.,2010	Stationary, Walk, Bike, Motorcycle, Car, Bus, Metro, Train	Gps Accel	1s (gps) 0.04s (accel)	4	1	**1**	82.1
Reddy et al.,2010	Stationary, Walk, Run, Bike, Vehicle	Gps Accel	1s (gps) 0.03s (accel)	16	4	**1**	93.7
Stenneth et al.,2011	Stationary, Walk, Bike, Car, Bus, Train	Gps Gis	14s (gps)	6	7	**3**	92.8
Hemminki et al.,2013	Stationary, Walk, Car, Bus, Train, Metro, Tram	Accel	0.01s (accel)	16	27 +5	**3**	84.9
Prelipcean et al., 2017	Walk, Bike, Car, Bus, Metro, Train, Ferry	Gps Accel	50m (gps) 0.2s (accel)	9	11	-	90.8
Yu et al.,2014	Stationary, Walk, Run, Bike, Vehicle (Motorcycle, Car, Bus, Metro, Rail, Train)	Accel	0.03s (accel)	224	22 +8	**1**	91.5
Yanyun et al.,2017	Train, Metro, Bus, Car	Accel	0.01s (accel)	30	169	**1**	98
Ashqar et al.,2018	Car, Bus, Bike, Run, Walk	Gps Accel	0.04s (gps) 0.01s (accel	10	80	**2**	97

B. RESEARCH AIMS AND ARTICLE ORGANIZATION

The aim of our research has been to realize a solution overcoming the state-of-the-art solutions to classify the transportation modes to deliver personalized services for:

- sustainable mobility, to incentivize ecological transportation choices, suggesting alternative public mean of transport (bus/tram) instead of the private car/motorbike.
- healthy suggestions, better and enjoyable life, to stimulate users in dedicating a part of their time and moving needs to exercising their body.
- implementing city strategies to change city user attitudes [Badii et al., 2017b], [Badii et al., 2018].

With this pourpose, the real-time identification of a private transportation mode (car or motorbike) has a central role in assistance messages delivery. Therefore, according to the above described real operating conditions, the techniques have to produce high classifications accuracy to identify transportation modality of a user, in the presence of (i) large discontinuities samples of data (from sensors and sporadic communications to the central computation modules), (ii) relevant differences which may be due to the different kind of mobile phone features in terms of sensors and precision.

Therefore, the proposed solution overcomes the above-mentioned solutions at the state of the art, for the aspects focused on sensor energy consumption factors and real conditions. The solution has been tested on a real application (delivered to the users via official App stores such as Google Play Store, Apple App Store, and accepted by common users, see "*Tuscany where what….*" on the stores). As described in the following, it is capable to cope with the constrains introduced by terminal manufactures on battery usage for background and foreground services. Moreover, no restrictions on the modality of mobile device usage have been imposed, differently to what has been imposed in the state of the art experiments in which the devices have been asked to keep: (i) the application running in foreground to get more precise GPS data, (ii) the device in a proper position/ orientation during the usage; and/or to (iii) use specific devices.

The paper is focused on the classification of the transportation mode/means whether an individual is: (i) stationary, (ii) walking, (iii) on a motorized private transport (car or motorbike), or (iv) in a public transport (tram, bus or train). The classification model proposed has been produced by using open and real-time data of Sii-Mobility/Km4City project and infrastructure (which is national smart city project of Italian Ministry of Research for terrestrial mobility and transport, http://www.sii-mobility.org). Sii-Mobility is based on Km4City data aggregation and analytics infrastructure (https://www.km4city.org) in the Tuscany area, Italy, and its Smart City solutions. Research results have been produced with the aim of defining solutions for sustainable mobility, stimulating citizens towards virtuous behaviors, providing info mobility, etc. The research project conducted a large experimentation of the solution with the support of Public Transport Operators: BUSITALIA, CCTNORD.

This paper has been organized as described in the following. Section II reports the general architecture for data collection, from devices to server, and related data analytics. In Section III, a list of the identified metrics is reported, mainly related to: baseline, GPS, accelerometer, and historical data. Section IV proposes a comparison of predictive models exploiting the collected data from Km4City, to arrive at identifying the best resulting approach in terms of classification precision and recall. Conclusions are drawn in Section VI.

II. ARCHITECTURE AND DATA COLLECTION OVERVIEW

The proposed solution relies on a client-server architecture, where the mobile application can be installed on different operating systems with different versions [Badii et al., 2017a]. The sensors' values collected on the mobile device (client-side) are sent to the server that enriches them with additional context information derived data (GIS, geographical information system and knowledge), etc., as described in the sequel. At the same time, the server executes the real time classification algorithm to compute the transportation mean classification for each user. The information is stored on server as a report on the preferred user's travel mean. On this basis, strategies triggered when the user behavior reaches certain specific conditions has been activated – i.e., to assist and/or engage the users in their daily activities (even rewarding them, in the cases of virtuous behavior). For example, a strategy for stimulating the city users may be based on a firing condition which sends a suggestion to all city users that take their private car to perform the same trip path at least 3 times per week, and at the same time the trip could be easily performed by using public transportation. Thus, the system may inform those city users of the possible alternative, and some of them may follow the suggestion. As a result, by exploiting the user behavior analysis, the solution may detect the acceptance of the suggestion by detecting of change of behavior and may automatically reward the user with a bonus or discount, and deliver congratulations. See [Badii et al., 2017b], on rules and strategies. Figure 1 provides a high-level overview of the software architecture and its main components.

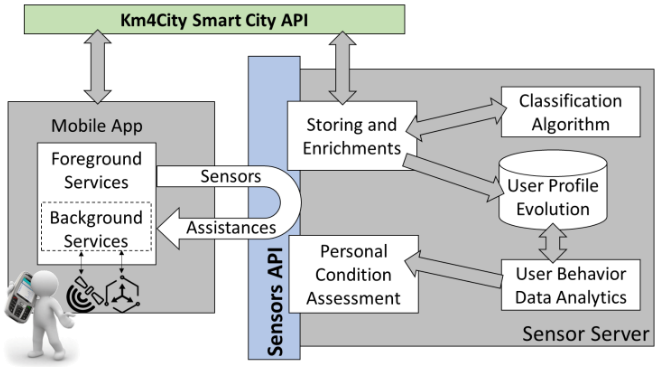

Figure 1. System architecture.

III. DATA AND FEATURES DEFINITIONS

The features considered by the classification algorithm have been selected from a larger set considered during the preliminary analysis and experiments. The process of features reduction has been performed by assessing their relevance in the classification. The aim was to identify the smallest subset of features without reducing significantly the precision of the travel mean's classifications. As a result, **Table 2** includes the selected metrics and the features, classified in 4 categories, collected from the mobile as **Sensor Data Package**, with those computed from the server-side to be used by the classification algorithm. Some of these features can be used for both users' traveling mean classification, and for creating firing conditions for implementing strategies. In **Table 2** "Where" can be: "D" when the measure is produced on the Device, and "S" when is computed on server-side. Each measure is collected/referred at a given **Day and Time**, and from this value can be easily derived from the device or from server if the day is a working day or not (**Non-Working Day**). The same approach can be followed to detecting the **Time Slot** in which the measure has been collected. The Time Slot strongly influences the attitude of the city users to move by using different means.

Table 2. Overview of Sensor Data Package feature measured at a given time from the mobile or computed on server-side.

categories	Metrics	Description of metric variable	Where
Day/Time Baseline and GPS	Day and Time	Day and Time of the sample package	D
	Non-Work-ing day	1 if weekend or vacation, 0 if it is a working day	D/S
	Time Slot	Slot of the day (morning, afternoon, evening, night)	S
	GPS latitude and longitude	Position of the device in GPS coordinates	D
	Accuracy	GPS Sensor's Accuracy from the mobile device	D
	Location Measure kind	Types of Location measure: GPS, Network, Mixed/Fused	D
	Speed	Speed as provided by the GPS driver of the mobile (as m/s)	D/S
	Average Speed	Average speed of the measures collected in the last two minutes	D/S
	Phone Year	Year/age of the terminal	D
	BDS	Availability of a BDS compliant	D

		GPS Sensor	
	User Type	User Type: commuter, citizen, students, tourist, e tc.	D/S
Accelerometer	Average linear magnitude of ac celeration	Average of the acceleration magnitude calculate on five measurements	D
	Linear accelerat ion of X-axis	Acceleration of the device along the X-axis, purged by Earth gravity	D
	Linear accelerat ion of Y-axis	Acceleration of the terminal along the Y-axis, purged by Earth gravity	D
	Linear accelerat ion of Z-axis	Acceleration of the terminal along the Z-axis, purged by Earth gravity	D
Proximity	Rail Line	Bool indicating if the device is in proximity of a rail line	S
	Sport Facilities	Bool indicating if the device is in proximity of a sport facilities	S
	Tourist Trail	Bool indicating if the device is in proximity of a tourist trail	S
	Green Areas	Bool indicating if the device is in proximity of a green areas	S
	Bus/ Light-rail Line	Bool indicating if the device is in proximity of a bus line or a light-trail line	S
	Cycle Paths	Bool indicating if the device is in proximity of a cycle path	S
Temporal window	Previous speed	Speed of the device of the previous 12 minutes	S
	Previous averag e speed	Average speed on the measures collected in a 12 minutes time slot	S
	Previous media n speed	Median speed on the measures collected in a 12 minutes time slot	S
	Speed distance	Speed (m/s) calculated on the distance between t wo consecutive coordinates and the time passed between the observations	S

As described in Section II, the information about the user's movements is collected from the device sensors. If the user has the mobile application in foreground, the data are sent to the server every 1 minute and 30 seconds (sending interval). This interval can be reduced by the user (via the setting of the App) to an update up to 30 seconds, to have a more accurate assistance. If the App is not used, the data collection is performed in background modality, thus the measures and sending rates may become up to 3/5 minutes, forced by the operating system/device, which in some cases can hibernate the App. Therefore, in order to make the solution viable in real conditions (differently from the state-of-the-art solutions), a set of strategies and robust classification algorithms have been put in place. Among them, solutions for filtering noise and GPS errors, and for smoothing the sequence of the user locations (user trajectory) have been used.

A **Sensor Data Package** l_i represents the user context at a specific time t_i and is composed by the **GPS latitude and longitude** (according to a Location Measure kind), speed, and accuracy of the measure plus a list of N additional features (*feat-1…feat-n*):

$$l_i = \{latitude_i, longitude_i, speed_i, accuracy_i, feat\text{-}1_i, \dots, feat\text{-}n_i\}$$

A user trajectory t_{ir} is a sequence of l_i that describes the movements of a user to move from l_i to l_r:

$$t_{ir} = \{l_i, \dots, l_r\}$$

A segment s_{uv} is a trajectory t_{uv} in t_{ir} where a user keeps the same mobility mean:

$$l_i \rightarrow mobility\text{-}A \rightarrow \underbrace{l_u \rightarrow mobility\text{-}B \rightarrow l_v}_{t_{uv}} \rightarrow mobility\text{-}C \rightarrow l_r$$

The distance between l_u and l_v can be approximated by using flat-surface formulae between the two coordinates $<latitude_u, longitude_u>$ and $<latitude_v, longitude_v>$.

A measure of the terminal **Speed** can be directly retrieved from the GPS sensor (for example, every 30 seconds or at the rate imposed by the device). On the other hand, the above-mentioned Average Speed of **Table 2** is calculated over the sequence of l_i in the same sending slot from the mobile device, to cut out eventual errors coming from GPS sensor. If the mobile App is in foreground the **Average Speed** is computed every 2 minutes (4 measures of 30s, if any). If the mobile App service for collecting data is in background, and 2 minutes passed before a measure is available (probably the operating system put the application in hibernate mode). The service tries to wake up whenever it is possible (if the operating system on the device allows us to wake the service up), to retrieve a bounce of new l_i to calculate a more precise Average Speed. **Figure 2** overviews the scenario.

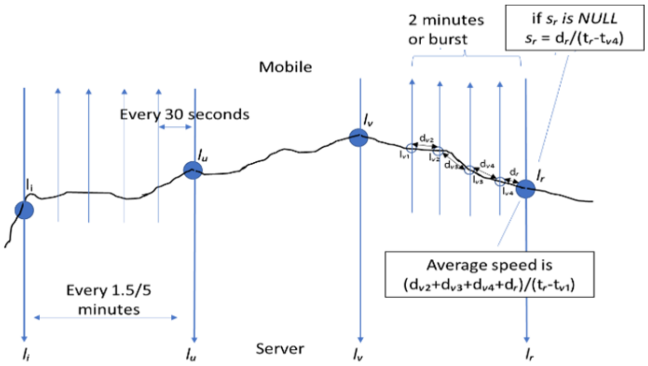

Figure 2. Speed and average speed.

The **Location Measure kind** is an important feature to understand the location measures reliability. Usually, the measures obtained and marked as "GPS" by the mobile device are quite accurate, even if they suffer time by time of well-known problem of shading (e.g., urban canyoning) or blocked (under the bridge) [Misra et al., 2006]. The location measures, labeled as "Network", resume the position from the location of the available Wi-Fi hotspots or GSM/4G/5G in the mobile connection; while those marked as "Mixed" modality is obtained by the operating system by merging the previous strategies according to different algorithms that may depend on the operating system kind, sensor kind, etc. The Location Measure kind strongly depends on the factory settings of the device, that make very difficult to force a pre-determinate modality from the App. The **Accuracy** of the GPS measure is reported in meters from the device and can be used from the classificatory algorithm to eventually discharge entries. Terminal model and its characteristics are also tracked and passed to the classification algorithm. Thus, the **Phone Year** of production of the device and the characteristics of the GPS sensors strongly influence the reliability of measure and thus have and have been considered as variable, differently from the state-of-the-art solutions. Old terminals usually support just A-GPS modality, meanwhile new ones' support also GLONASS and BDS standards.

A. ACCELEROMETER FEATURES

Values from the **Accelerometers** of the terminal/device are always available and are sampled. Using the linear acceleration of the device avoids taking measures influenced by device orientation (horizontal or vertical, in the hand or in the pocket). Not all the mobile devices provide this information (some of them just return the non-linear values, influenced by the gravitational acceleration, and orientation, thus needing a de-rotation). On the other hand, almost all the relatively new devices already have this aggregated measurement available. **Phone Year** variable allows us to take this into account. Thus, the three measures of linear acceleration on three axes have been considered aggregating five consecutive acceleration measures for computing an average magnitude as:

$$\textit{Average Linear Magnitude of Acc} = \sum_{k=1}^{5} \frac{\sqrt{acc_{x_k}^2+acc_{y_k}^2+acc_{z_k}^2}}{5}$$

B. DISTANCE FEATURES

On the server-side, the **Sensor Data Package** collected from the devices via the App are enriched by computing and, in most cases, exploiting the Km4City knowledge base of the City via Smart City API. This allows to retrieve contextual information about the closeness of the device/user with respect to: Railway Line, Sport Facilities, Tourist Trail, Green Areas, Bus/Light-rail Line, and Cycle Paths. The closeness features are binary values that specify if the location is closer to those structures, in the range of 30mt. This derived information is very valuable for understanding some transportation means. For example, to be close to a Rail and/or Bus/Light-rail line for a number of points of a trip permits to infer bus/train modality (train, bus and light-rail run just in their closeness) with a very high probability. On the other hand, the closeness to a cycle path cannot directly infer that a user is using a bike because the user can be in its proximity by a car and with similar speed or a bike can run also away for the cycling path (see **Figure 3**).

Besides having instantaneous measurements about device/user's mobility, the speed values in the last 12 minutes time-frame is also computer on server-side, as well as the average and mean value between these measurements. This allows to reduce the noise overcoming disruptive mobility conditions mainly related to traffic congestion or temporary signal absence. This is also due to the fact that the service for collecting data on the mobile device runs on a real application (foreground/background) conforming to the policy of "energy saving" of the user to have shortage of data for up to 3/5 minutes. So that, in real conditions, it is very important to avoid battery drainage warning, that may stimulate the user to un-install the App from the device. In order to perform an addition refinement on speed measures, mean and median speed and distance between GPS coordinates are also computed.

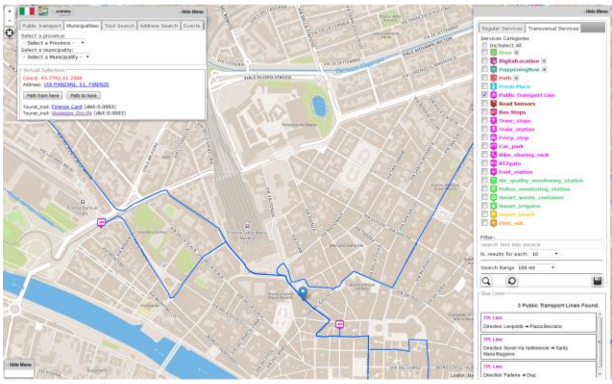

Figure 3. Bus-line in proximity computation

C. TEMPORAL WINDOW FEATURES

The **User Type** specified by the user in the App during installation or setup permits contribute to the classifications and to the strategies. The User Types are: citizen, commuter, student, tourist, etc. We noticed that different profiles present a different approach in everyday mobility and, so on the transportation mode they normally use.

IV. RESULTS FROM CLASSIFICATION/PREDICTION MODELS

According to the above described data, the challenge was to predict the transportation mode, whether an individual is stationary, or is walking, or moving on a motorized private transport (car or motorbike) or using a public transport (tram, bus or train). The experiment has been conducted on about 30.000 observations, collected from April to August on **38 different users and 30 different kinds of devices.** Note that, each user can use the mean of transport they want. When the mode of transport is changed, the user was asked to notify the change to the App for creating the learning set and for validation. As mentioned above, no restriction was imposed on how the phone should be held during movement (foreground/background, on hand or bag, etc.). Unlike the experiments reported in the literature, most of the data was collected in the background because the phones were kept in pocket or bag, in fact there is a non-conformity in the frequency distribution of the collected data. In details, the frequency average is equal to 180 seconds and the variance is equal to 13240 seconds. The frequency distribution of the sampling period is reported in **Figure 4**.

The training set has been created by randomly selecting the 80% of the collected data, while the test set was the remaining 20%.

In the general framework, three different approaches were more successfully considered -- i.e., Random Forest (RF), Extremely Randomized Trees (Extra-Trees), and the

Extreme Gradient Boosting procedure (XGBoost). Those approaches have been tested by using the above presented features/metrics (see **Table 2**), classified by categories as: *baseline* and *GPS* features, *accelerometer* features, *distance* features and *temporal window* features. The comparison among those models has been reported in **Table 3**, in terms of resulting data. From the comparison, it is evident that all the approaches are capable to produce satisfactory predictions (the accuracy for each model exceeds 90%) for the identification of the transportation means.

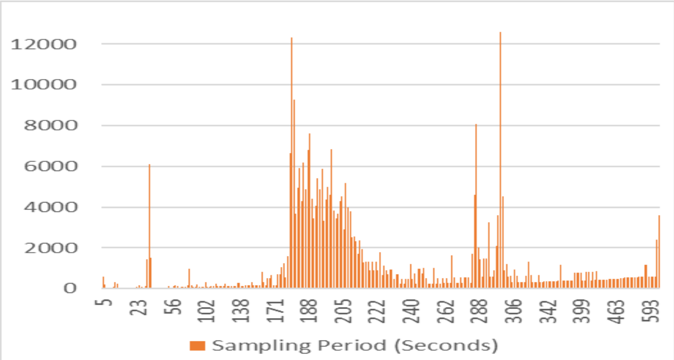

Figure 4. Frequency Distribution of Sampling Period.

According to the data results of **Table 3**, the differences among the different approaches provide the evidence that the **Extra-Trees** resulted to be the better-ranked approach in terms of accuracy and F_1 score. In **Table 3**, the F_1 score is reported: F_1 score has been used to measure the models' performances. This is a measure to evaluate the robustness of a model for making predictions, as a compromise between precision and recall:

$$F_1\ score = 2 \times \frac{Precision\ \times\ Recall}{Precision\ +\ Recall}$$

$$Precision = \frac{\#\ \text{correctly classified istances into class}\ i}{\#\ \text{istances classified as class}\ i},$$

$$Recall = \frac{\#\ \text{correctly classified istances into class}\ i}{\#\ \text{istances belonging to the class}\ i},$$

Table 3. Classification Models Comparison on four classes of transport mode: stationary, non-motorized, private transport, public transport.

Classifier Models	Accuracy	Precision	Recall	F_1 score
Extreme Gradient Boosting	0.947	0.773	0.828	0.800
Random Forest	0.942	0.774	0.869	0.819
Extra-Trees	**0.953**	**0.827**	**0.869**	**0.847**

According to this our first result, the Extra-Trees algorithm achieves an accuracy of 0.953, and a precision of 0.827. It should be remarked that, these results have been obtained and can be produced by observing data coming from a large range of devices and a variable sampling rate (up to 5 minutes). The model produce allows to understand if a user is moving with a public or private transport.

On the contrary, in [Reddy et al., 2010], a precision of 0.937 has been obtained by using a single device, Nokia n95, and a constant sampling rate of 60s, which is not realistic with present mobile operating systems. With Reddy's classification was only possible to know if a user is moving with a motorized vehicle. The same considerations apply to: [Stenneth et al., 2011] where data come from three different devices and they are taken with a constant rate of 15s achieving a precision of 93.7%; and to [Yu et al., 2014] achieving a precision of 91% with accelerometer sensor data only, without distinguishing the type of motorized transport.

Moreover, Table 4 reports the assessment of the results performed for each traveling mean classification for the Extra Tree procedure according to our first result. The traveling mean class with lower accuracy is Walk. This is probably due to the fact that, it is not easily to understand if a user is walking or not, since the GPS sensors accuracy is very noisy in indoor scenarios, with frequent jumps passing from the different modalities: wifi- mixed, etc.

Table 4. Extra-Trees Prediction Model: Statistic by class.

Extra Trees Model	Stay	Walk	Private Transport	Public Transport
Sensitivity	0.978	0.731	0.869	0.917
Specificity	0.901	0.988	0.987	0.996
Pos Pred Value	0.977	0.770	0.827	0.936
Neg Pred Value	0.904	0.985	0.990	0.994
Balanced Accuracy	0.940	0.859	0.928	0.956

We also tested the effect of combining the solution with a SuperLearner approach without obtaining better results.

A. ASSESSING THE INFLUENCE OF FEATURES

A comparison in terms of accuracy, precision and recall of the Extra-Trees multi-class approach has been computed considering four combinations of the different categories of data (as reported in **Table 2**):

- baseline features and distance feature;
- baseline, distance feature and accelerometer features;
- baseline, distance feature and temporal window features;
- baseline, distance, accelerometer, temporal features together. (**Full Model**)

This set of combinations of feature categories permits to assess the flexibility of our approach in real operative conditions, where a variety of devices have to be supported, since not all devices support the full combination of categories. The results obtained by using different subsects of feature categories are reported in **Table 5.** Please note that the differences among the different cases for feature categories are substantial. The results suggest that the best choice in terms of precision is still the usage of model exploiting all the categories together, thus demonstrating that the model is flexible and resilient with respect to the device kind. Please note that the Boolean value detecting a close transportation line (i.e., proximity feature in the table) improves the classification effectiveness: the accuracy passed from 0.91 to 0.92 and higher.

Table 5. Extra Tree Model results on four classes of transport modality (stationary, non-motorized, private transport, public transport) considering four combinations of the different features.

Model features categories	Extra Tree Model results			
	Accuracy	Precision	Recall	F_1 Score
Baseline and GPS	0.910	0.682	0.751	0.714
Baseline and GPS + proxi mity	0.924	0.739	0.691	0.715
Baseline and GPS + proxi mity + Accelerometer	0.926	0.814	0.744	0.777
Baseline and GPS + proxi mity + Temp window	0.949	0.805	0.787	0.787
Baseline and GPS + proxi mity + Accelerometer + Temporal window	**0.953**	**0.827**	**0.869**	**0.847**

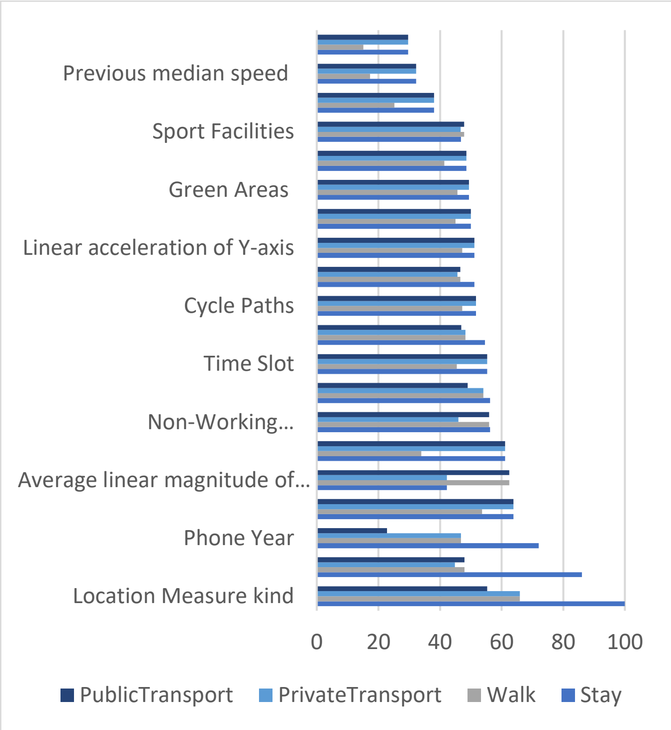

Figure 5. Variables Importance across the classes of the Extra-Trees full model.

In **Figure 5,** the features listed in **Table 2** are reported in order of importance across the classes for the prediction of the Extra-Trees Full Model, (the model with all the categories of covariates).

B. REAL CONDITION SCENARIO VS HIERARCHICAL APPROACH LIMITATIONS

Several considerations have been already presented about the critical aspect of working on real operating conditions. Battery drainage and the opportunity to support a contextual service for the users, even with the application in background mode, drove our research, despite little decrease of accuracy and precision. We decided to design a client-server architecture to support a finer classification, using GIS data easier available on the server side (avoiding user terminal network bandwidth usage to eventually download from remote) and to support technologies to aggregate information cross-terminal and user agnostic. Implementing a central server-side classification algorithm leaves open also the chance to auto-update scenario with feedbacks provided directly by the user. However, a real condition scenario can be affected by some limitations that cannot be solved either if a hierarchical approach is applied. This is due to the fact that the phone/user characteristics can be manifold, e.g., the presence of accelerometer information, the different type/generation of gps sensor, the presence of information related to the temporal window, etc. For this reason, the classification model has to be flexible and the training dataset has to be as much as possible various (e.g., any kind of generations, manufactures, years, characteristics, etc.) without any restriction. The application of a two-steps approach may lead to a loss of accuracy due to a loss of information and can be more time consuming in terms of execution time and number of different training models. In detail, during the second step, six different training models have to be executed, one for each combination between pairs of the transportation modes (selected during the step-one), considering that the classes of transportation means are four. In addition, a specific model has to be created depending on the characteristics of the device and of the users, considering four combinations of the different categories of data (reported in Table 2).

A two-steps hierarchical approach has been proposed. In the first step a multi-class classifier algorithm has been adopted to classify the transportation modalities. After the first classification, the classes with a probability lower than a threshold of 0.90 ($prob$ < 0.90) have been re-classified in the second step, while the classes that have a probability higher than 0.90 are considered as correct and excluded from the re-classification test set. During the second step six different binary classification model have been trained, one for each pair of transportation modality.

On the other hand, a single step classification model has been presented and different models have been compared. The Extra-Tree algorithm can be considered as the best and final solution: it was found to produce the best performance in terms of average accuracy (**0.953**) and time consuming. In detail, four different models have been trained to make the approach as flexible as possible. The necessity of this flexibility is because the solution has to be applied in a real condition scenario, for different phone/user characteristics, in any pseudo real-time context.

V. CONCLUSIONS

This research has been focused on presenting a solution to create a classification system that uses mobile devices' sensor values and GIS data (user contextual information) to identify the transportation mean of users: stationary, walking, on a motorized private transport (car or motorbike) or in a public transport (tram, bus or train). The goal has been to define a solution for sustainable mobility, delivering to the user useful personalized assistance messages. A number of metrics and features have been chosen as the *baseline and GPS*, the *distance*, the *accelerometer* data and the *temporal windows* data. The research documented in this paper demonstrated that a one-step multi-class classifier solution was found to produce the best performance in terms of average accuracy and time consuming if compared to a hierarchical approach. In detail, the Extremely Randomized Trees exploiting all the discussed above data can be a robust approach for reliable, precise and fast estimation of transportation means. The proposed solution overcome those of the literature since it presents a solution that is capable to produce reliable results in real conditions (i.e., real-time applications and background modality of operations) with a real set of devices and in particular: (i) addressing a large number of devices providing different features, different GPS sensors, different accelerometer sensors, etc., (ii) working with time variable samples of the data that may be due to the different operating systems, energy saving setting, etc., which are not under control of the App and thus are a strong constraint to realize real applications, background/foreground modality of operation; (iii) exploiting a number of different features and obtaining results with higher precision and accuracy. For these reasons, features related to the type of phone, e.g., the presence of accelerometer, phone year, location provider etc., have been considered in the prediction model, contributing to perform corrections in the model. The prediction model proposed has been created by exploiting open and real-time data of the Sii-Mobility (national smart city project of Italian Ministry of Research for terrestrial mobility and transport, http://www.sii-mobility.org). Sii-Mobility is un turn based on Km4City infrastructure http://www.km4city.org active in the Florence area, Italy since 2015. The solution presented has been deployed as an additional feature on Smart City Apps in the Tuscany and Florence areas for sustainable mobility, which is now in place for stimulating the private mover toward a more sustainable mobility with the collaboration of three major public transportation operators: ATAF, BUSITALIA and CTTNORD. Most of the computations were conducted in R Statistical Environment (https://www.R-project.org/), and then implemented in real time. In addition, the same solution has been used in Snap4City mobile Apps with experiments performed in Antwerp and Helsinki on Android mobile Apps that are on Google Play. In those cases, the collection of data from the mobile have been authorized thanks to the signed consent according to the GDPR of Snap4City [Badii, et al., 2018b].

VI. Acknowledgements

The authors would like to thank the MIUR Smart City national founding SCN_00112 and to the University of Florence and companies involved for co-founding. Snap4City has been founded by Select4Cities PCP of the European Commission. Snap4City and Km4City are 100% open technologies and research of DISIT Lab. Thanks to the collaboration of three major public transportation operators in Italy for their support in putting the solution in operative conditions: ATAF, BUSITALIA and CTTNORD.

VII. References

[Ashqar et al., 2018] Ashqar, H. I., Almannaa, M. H., Elhenawy, M., Rakha, H. A., & House, L. (2018). Smartphone Transportation Mode Recognition Using a Hierarchical Machine Learning Classifier and Pooled Features From Time and Frequency Domains. IEEE Trans. on Intelligent Transportation Systems.

[Badii et al., 2017b] C. Badii, P. Bellini, D. Cenni, A. Difino, P. Nesi, M. Paolucci, "User Engagement Engine for Smart City Strategies", 3rd IEEE Int. Conf. on Smart Computing, 2017.

[Badii et al., 2018] C. Badii, P. Nesi, I. Paoli, "Predicting available parking slots on critical and regular services exploiting a range of open data", IEEE Access, 2018

[Badii, et al., 2018b] C. Badii, et al., "Snap4City: A Scalable IOT/IOE Platform for Developing Smart City Applications", Int. Conf. IEEE Smart City Innovation, Cina 2018, IEEE Press. https://ieeexplore.ieee.org/document/8560331/

[Biancat et al., 2014] Biancat, Jacopo, Chiara Brighenti, and Attilio Brighenti. "Review of Transportation Mode Detection techniques." ICST Trans. Ambient Systems 1, no. 4 (2014): e7.

[Biljecki et al., 2013] Biljecki, Filip, Hugo Ledoux, and Peter Van Oosterom. "Transportation mode-based segmentation and classification of movement trajectories." International Journal of Geographical Information Science 27.2 (2013): 385-407.

[Hemminki et al., 2013] Hemminki, S., Nurmi, P., & Tarkoma, S. (2013, November). Accelerometer-based transportation mode detection on smartphones. In Proc. of the 11th ACM Conf. on Embedded Networked Sensor Systems (p. 13). ACM.

[Lv et al., 2018] Lv, Zhihan, et al. "Government affairs service platform for smart city." Future Generation Computer Systems 81 (2018): 443-451.

[Manzoni et al., 2010] Manzoni, Vincenzo, et al. "Transportation mode identification and real-time CO2 emission estimation using smartphones." SENSEable City Lab, Massachusetts Institute of Technology, nd (2010).

[Misra et al., 2006] Misra, Pratap, and Per Enge. "Global Positioning System: signals, measurements and performance second edition." Massachusetts: Ganga-Jamuna Press (2006).

[Prelipcean et al., 2017] Prelipcean, Adrian C., Gyözö Gidófalvi, and Yusak O. Susilo. "Transportation mode detection–an in-depth review of applicability and reliability." Transport Reviews 37.4 (2017): 442-464.

[Reddy et al., 2010] Reddy, Sasank, et al. "Using mobile phones to determine transportation modes." ACM Transactions on Sensor Networks (TOSN) 6.2 (2010): 13.

[Stenneth et al., 2011] Stenneth, Leon, et al. "Transportation mode detection using mobile phones and GIS information." Proc. of the 19th ACM SIGSPATIAL International Conf. on Advances in Geographic Information Systems. ACM, 2011.

[Wang et al., 2010] Wang, Shuangquan, Canfeng Chen, and Jian Ma. "Accelerometer based transportation mode recognition on mobile phones." Wearable Computing Systems (APWCS), 2010 Asia-Pacific Conference on. IEEE, 2010.

[Yanyun et al., 2017] Yanyun, G., Fang, Z., Shaomeng, C., & Haiyong, L. (2017, September). A convolutional neural networks based transportation mode identification algorithm. In Indoor Positioning and Indoor Navigation (IPIN), 2017 International Conference on (pp. 1-7). IEEE.

[Yu et al., 2014] Yu, Meng-Chieh, et al. "Big data small footprint: the design of a low-power classifier for detecting transportation modes." Proc. of the VLDB Endowment 7.13 (2014): 1429-1440.