Semantic Labeling of Places based on Phone Usage Features using Supervised Learning

Semantic Labeling of Places based on Phone Usage Features using Supervised Learning

Alejandro Rivero-Rodriguez, Helena Leppäkoski, Robert Piché
Tampere University of Technology
Tampere, Finland
{Alejandro.Rivero, Helena.Leppakoski, Robert.Piche}@tut.fi

***Abstract*—Nowadays mobile applications demand higher context awareness. The applications aim to understand the user's context (e.g., home or at work) and provide services tailored to the users. The algorithms responsible for inferring the user's context are the so-called context inference algorithms, the place detection being a particular case. Our hypothesis is that people use mobile phones differently when they are located in different places (e.g. longer calls at home than at work). Therefore, the usage of the mobile phones could be an indicator of the users' current context. The objective of the work is to develop a system that can estimate the user's place label (home, work, etc.), based on phone usage.**

As training and validation set, we use a database containing phone usage information of 200 users over several months including phone call and SMS logs, multimedia usage, accelerometer, GPS, network information and system information. The data was split into visits, i.e., periods of uninterrupted time that the user has been in a certain place (Home, Work, Leisure, etc.). The data include information about the phone usage during the visits, and the semantic label of the place visited (Home, Work, etc.). We consider two approaches to represent this data: the first approach (so-called visits approach) saves each visit separately; the second approach (so-called places approach) combines all visits of one user to a certain place and creates place-specific information. For place detection, we used five popular classification methods, Naïve Bayes, Decision Tree, Bagged Tree, Neural Network and K-Nearest Neighbors, in both representation approaches. We evaluated their classification rates and found that: 1) Bagged Tree outperforms the other methods; 2) the places data-representation gives better results than the visits data-representation.

Keywords—Location and positioning services, Context Inference, Place detection, Semantic positioning

I. Introduction

The use of smartphones has dramatically changed during the last decade. Firstly, whereas only 1 % of worldwide population owned a smartphone in 2006, by the end of 2012 the number reached 22 % [1]. Secondly, mobile technology has developed extraordinarily and the most well-known smartphone vendors (e.g. iOS, Android, WP) made available interfaces that offer possibilities for third-parties to develop specific-purpose applications. This, together with the inclusion of inexpensive physical sensors, encouraged developers to use users' information and build context-aware applications. The most well-known case of context-aware application is the so-called location-based services [2].

All the aforementioned developments have had an impact on how people use smartphones. We seem to be far from the era when phones were used exclusively for calling and sending text messages. Besides these, they are currently used for a variety of activities such as playing games, web browsing, e-mail, internet based messaging, communication and social media, taking photographs, recording or watching videos, and using specific-purpose applications. Therefore, users demand (smarter) context-aware applications that are adequate to their needs. For instance, the personal assistant application Google Now infers the locations of your Home and Workplace by tracking your movements. With such information it provides you with valuable information, for instance suggesting the best route from your current position, according to current traffic conditions [3].

To achieve such goals, the most typical approach is to use physical sensor information exclusively. However, we can use other useful information to infer context, such as phone usage (e.g. phone calls, battery status) and third application data (e.g. calendars, Facebook status). It should be noted that contextual information must be used according to the laws and regulations that define the requirements for privacy protection. Matching these requirements is compulsory for applying these methods in real-world scenarios.

In this work we use the so-called MDC database [4], where about 200 users used Nokia N95 devices normally for between 3 and 18 months. All the information of the usage of the phones was automatically collected and anonymized. The data includes the logs of phone calls and SMS, calendar entries, multimedia displayed, GPS information when available, network information and system information (e.g. battery status, device inactive time). After the data collection, a clustering algorithm was used to identify the most relevant places for each user, who were then asked to label them manually [5].

Using the aforementioned data, we use supervised learning methods to create a place detection algorithm that estimates the semantic label of the current place based on the phone's current usage features.

The rest of this article is organized as follows: in Section 2 we outline the background of our work, highlighting the current needs for place detection. In Section 3 we present the data and the preprocessing used in this work. Section 4 describes the different classification methods and presents evaluation test results. Finally, in Section 5 we conclude the article.

II. Background

Research on context aware systems began in earnest in the early 1990's [6]. Context *can refer to any information that can be used to characterize the situation of an entity, where an entity can be a person, place, or physical or computational object* [7]. To infer a user's context, we use sensor information. According to Baldauf et al. [6], the notion of a sensor is typically generalized. We distinguish three types of sensors:

Physical sensors are the most widespread form of sensor. They are devices that detect and respond to some type of input from the physical environment and capture physical data.

Virtual sensors capture contextual information from applications and services. They can be based on local services (e.g. calendar) or external services (e.g. weather forecast).

Logical sensors provide contextual information by combining information from physical and virtual sensors.

However, most existing systems consider the physical sensors [8], including the sensors related to the user's position, such as GPS, accelerometer, gyroscope (allowing e.g. activity recognition)[9],[10], or sensors that measure the properties of the user's environment, such as magnetic field, light, or properties of various radio signals around the user [11], [12]. Regarding virtual sensors, one of the most used is the user's language. For instance Google provides developers with the user's language through Google Developers API.

Some researchers point out that the usage of mobile phones can provide meaningful information about the user's context [13-16]. Reference [13] states that the user's context can be inferred based on the usage of applications (e.g., calls, e-mail, web browser).

In this work we investigate the main challenges and possible solutions for place detection, a particular case of semantic labeling. There are two reasons to focus on place detection. The first reason is the great value of this information, and its implications: Many context-aware applications can provide better services by using user specific information. This interests companies like Google or Microsoft. The second reason is the lack of good methods. By observing current products in the market, such a Google Now [3], one could think, that the problem is already solved. However, these methods are not yet accurate enough and new solutions are needed.

We apply different supervised learning methods on MDC data to find models that, based on the mobile phone usage patterns, allow assigning semantic labels to the places the user visits.

The goal of [14-16] is similar to ours, i.e., semantic place prediction, and they all use the data derived from the same database as the data in our work. However, they differ from our work in these aspects: the number of features we used for our classification method is only 14, while the other methods use more features; we use different sets of classifiers than the references, and we also present the comparison between the visits approach and the places approach.

III. Dataset description

In this section we describe the information contained in MDC database and identify the most relevant features for place detection.

The data used in this work is obtained from the MDC Database made available by Idiap Research Institute, Switzerland and owned by Nokia [17], [4]. The dataset contains Nokia N95 smart phones usage data, collected by nearly 200 users over time periods that for many users exceed one year [17]. From this database, we extracted the data that was collected during visits where the user stayed in the same place at least 20 minutes; these are defined in a database table that defines for more than 55 000 visits the start and end times of the visit, user id, and place id. The place labels for the place ids are defined in a separate table places.csv.

Based on these data, we queried from the database the following phone usage data for each visit, i.e., for a given user, all data entries between the start and end times of the visit:

System data, including battery and charging status and counter for inactive time

Call log, including durations of each phone call

Acceleration based activity data, including accelerometer based estimates of the user's motion mode: *idle/still*, *walk*, *car/bus/motorbike*, *train/metro/tram*, *run*, *bicycle*, or *skateboard*

From these data entries, we computed for each visit the features to be used in the classification task. We decided to use only such sensor data that can be assumed to be available also for a real time application on a phone without violating the privacy of the user. Our feature list includes the following:

duration duration of the visit in seconds

startHour time of the day when the visit started (0, 1, …, 23)

endHour time of the day when the visit ended (0, 1, …, 23)

nightStay proportion of the visit duration that is between 6 pm and 6 am

batteryAvg average battery level

chargingTimeRatio proportion of the visit duration when the charging has been on

sysActiveRatio proportion of the visit duration when the system has been active, i.e., inactive time did not grow

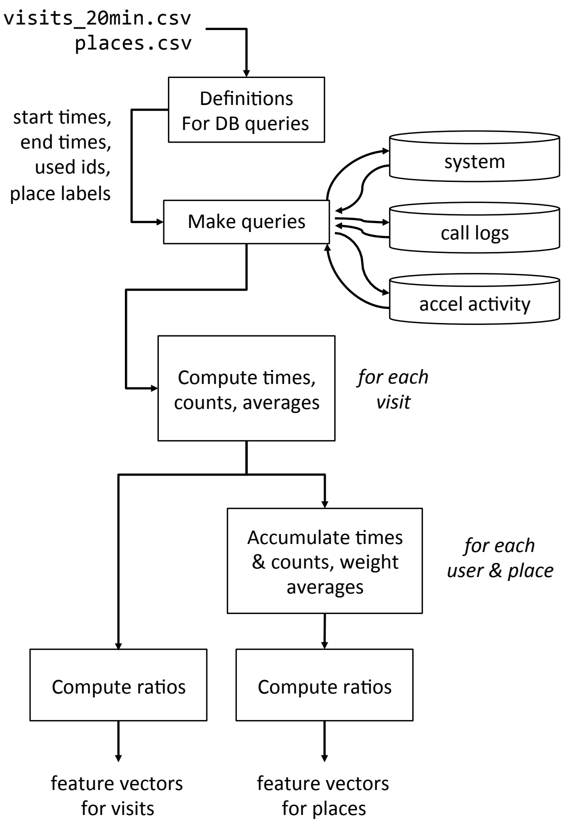

Figure 1. Data processing to obtain the features.

sysActStartsPerHour number of status changes from system inactive to system active divided by the visit duration in hours

For features related to calls, both incoming and outgoing voice calls are taken into account:

callsTimeRatio the ratio of accumulated duration of calls to the duration of the visit

callsPerHour number of calls divided by the visit duration in hours

The features related to accelerometer based motion mode detection were computed using the reported motion modes. However, as the report for one time instance may include several different modes and includes also their probabilities, we used the probabilities to weight the times for the motion modes:

idleStillRatio proportion of the visit duration when the status is *idle/still*

walkRatio proportion of the visit duration when the status is *walk*

vehicleRatio proportion of the visit duration when the status is either *car/bus/motorbike* or *train/metro/tram*

sportRatio proportion of the visit duration when the status is either *run*, *bicycle*, or *skateboard*

In addition to these 14 calculated features, we also saved the place label to be used in the training and testing of the models:

placeLabel three possible labels: *Home*, *Work*, or *Other* (the last includes all the generally less frequent places, such as friend's home, transportation, restaurant etc.)

The place labels were provided by users [5]. First, the data were collected and the relevant places for each user were clustered. In a later stage, users were shown all the places in a map and were asked to label these places. We only consider places labeled with certainty, and left out those places users were not sure about or users did not label.

In total, the visits data includes 55 932 labeled visits by 114 distinct users. From the visits 28 921 instances are to Home (52% of all visits), 21 697 instances to Work (38%), and 5 314 instances to Other places (10%).

IV. Methods

We consider two alternatives for the data-representation, visits-data representation and places-data representation, explained in the subsection *Data Representations.* Once the data is extracted from the database in both representation schemas, we consider five well known classification methods. Our goal is to determine which classification method and which data-representation approach is the best for the semantic labeling of places.

A. Data representations

We consider two different approaches to represent the data. The places approach uses the features computed for each visit as such, so that the data includes several samples of one user's visits to each of the user's places. That means that there is one tuple for each location-user-period. Therefore, a user visiting home 3 times add three tuples to the learning data. We extract 55 932 labeled visits by 114 users.

The visit approach combines all the visits of one user to one place as one summarized sample. That means that there is one tuple for each user-place, which is calculated combining all the visit tuple user-place-time. The idea is that different users use

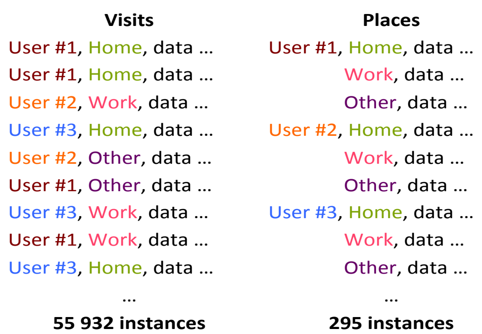

Figure 2. A schematic example illustrating the difference between visits and places data-representations.

their phones in similar ways in semantically similar places, for instance users use phone similarly at home. From the database we extract 295 labeled places by 114 users.

The difference between the approaches is illustrated in Figure 2. The data processing flow to obtain the features is shown in Figure 1. For instance, if a user visited home ten times in a week, the visit data-representation creates ten different data instances, while the place data-representation combines the ten visit data instances into one place data instance.

B. Classification Methods

In this work we test the following classification methods [18] using their implementations in the Statistics and Neural Networks toolboxes of Matlab. The classifiers learn using the training set, which is two thirds of the users in the dataset, a typical value used in Machine Learning.

Naïve Bayes (NB) is a pure statistical approach having an explicit underlying probability model, which provides a probability of being in each class rather than simply a classification. Naïve Bayes assumes that features are conditionally independent (to reduce computational cost), which works surprising well even if the independence assumption does not hold. There are no tuning parameters in this approach.

Decision Tree (DT) uses a machine learning approach which is generally taken to encompass automatic computing procedures based on logical or binary operations, in order to learn from a series of examples. This is probably the method that gives the most understandable results by humans, who can identify the most relevant features. For attribute selection we use Gini's diversity index. The features selected at the top of the three are the most relevant features for the classification. There are two options to avoid overfitting, pre-pruning and post-pruning. We chose post-pruning since pre-pruning requires determining when to stop growing the tree while building it, which is not an easy task. When the tree is built we post-prune the tree using Error Estimation. Intuitively, the method goes through the nodes of the tree comparing the original tree with the tree pruned on that node. The tree is pruned in that node if the pruned tree improves (or equals) the classification accuracy.

Bagged Tree (BT) combines different decision trees (with the same parameters as the decision tree above), each of which has been trained using different portions of the data. Using a voting system, each tree is given more weight in the region of the space where the classification rate is better. This method is proved to work better than single decision trees. We use ten decision trees, a typical value.

Neural Network (NN) is a brain-physiology inspired classifier. It consists of layers of interconnected nodes, each node producing a non-linear function of its input. The input to a node may come from other nodes or directly from the input data. Some nodes are identified with the output of the network. In particular, we used a Multi-layer perceptron with one hidden layer that contains ten hidden neurons. The decision of having these settings is based on the limited number of samples and the authors' experience. To train the network we used Levenberg-Marquardt optimization to update the weight and bias values.

K-Nearest Neighbors (KNN) is a statistical method that classifies an incoming instance according to the distance to the k nearest points in the training set. In our case, we set k=1 and search for the nearest neighbor, based on Euclidean distance. We selected k=1 because the computational cost is much lower. We also tested other values for k (3, 5 and 10), and the results worsen. For big datasets, this method can be prohibitive in CPU time. In general, it is not a good option if the classifier is in the user's device (e.g., mobile phone).

Another very important classifier in the literature is the Supported Vector Machine. The reason not to use this classifier is that it is basically a binary classifier, and we have to separate three classes. There are various heuristics to apply SVMs to multiple classifications, e.g., to construct two Support Vector Machines (e.g., classify *Home* or *Work-Other* and later classify *Work* or *Other*). However, we would need to make an a priori decision on what places are similar for the first classification.

Once we have built the classifiers based on the training data, we use the test data to evaluate the classifiers. The test set is the data corresponding to one third of the users, which has not been used previously to build the classifier. It is relevant to underline that the data has been split by users. Therefore a user's visit cannot be classified with the knowledge of other's visits, which is also more realistic. The test set is also labeled. Therefore, we have the information about the label (real values) of certain numbers of visits. For each visit, we ask our classifier whether the right label is Home, Work or Others. Then, we compare the real values with the predicted values by our classifiers. An accuracy of 53% means that 53% of the predicted values are equal to the real value.

V. RESULTS

Figure 3 shows the classification of each method using the visit-representation approach. All the methods but the Naïve Bayes have certain bias. They achieve high accuracy for the places *Home* and *Work*, and low accuracy for the place *Others*. The intuitive reason is that visits to *Home* or *Work* are more frequent than visit to place labeled as *Others*. Therefore the algorithms sacrifice accuracy in *Others* to achieve higher accuracy in *Home* or *Work*.

Figure 4 shows the same results using the place-representation approach. The difference of methods' accuracy is very small. This is probably due to high quality of the data representation, under the intuitive conjecture: if the data is very good, the selection of the method is not that relevant. There is no scientific justification for that high quality, but combining all the visits to one place may eliminate the visit-outliers. The disadvantages of place-representation methods are the

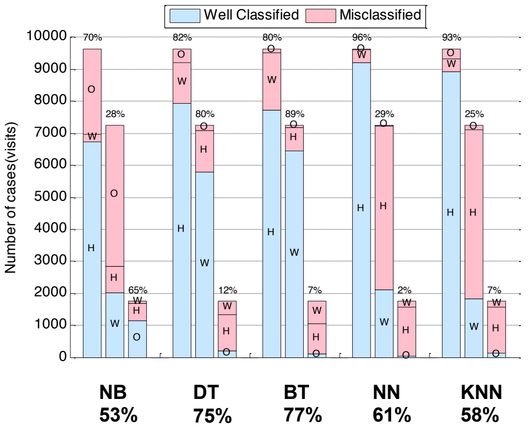

Figure 3. Classification rates (%) for different methods, using visits approach. The percentage of well-classified samples for each class is given above the bars. The overall percentage of well-classified samples for the classifiers is shown below the bars.

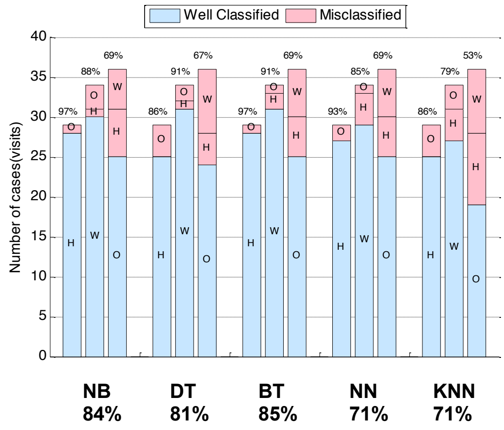

Figure 4. Classification rates (%) for different methods, using places approach. The percentage of well-classified samples for each class is given above the bars. The overall percentage of well-classified samples for the classifiers is shown below the bars.

following. First, it is more computationally expensive, because all the visits to places are calculated and they need to be combined, which requires extra computations. The second disadvantage is the so-called cold start problem, that the classification algorithm will not classify accurately the first places, until a certain number of visits to a place have been collected.

Therefore comparing the results of the methods using different data-representations, it is obvious that the places-approach provides higher accuracy, but it also has some restrictions. That implies there is not clearly a best option. It depends on the requirements of the problem to solve. However, the authors outline possible future research line, consisting of merging these two approaches. In others words, to utilize a classifier based on visits-data representation, or the other based on places-data representation, depending on the region of the space the point is located.

Comparing the results of different classifiers, the best algorithm seems to be Bagged Tree. However, the difference of accuracy with Naïve Bayes using the places approach is only 1%. This is not enough to statistically say that Bagged Tree is better and it might be conditioned by the portion of the data used for classification. On the other hand, the method KNN should be discarded because it does not offer improvements in accuracy while it has a high computational cost.

The best overall classification rates presented in [13-15] are in the range between 0.65 and 0.75. With our best classifiers, we achieve overall classification rates over 0.8.

In addition, we can see some of the relevant features by looking at the single decision trees in a top-down manner. These features that are chosen as split criteria in an earlier stage will be more significant to estimate the semantic place. These features are listed in descending order according to the relevance: night stay, stay duration, start time, battery status and idle still. One future improvement is the inclusion of the random forest methods, which is a similar method to the bagged trees. Even if it does not offer great improvements in accuracy, its results are more human-understandable, and the ranking of the most relevant features can be extracted directly.

VI. Conclusions

The test results indicate that places data-representation gives higher classification rates than visits data-representation. However, it should be noted that the places approach requires more processing work, with the consequent effect in computational costs as well as the cold start problem. In addition, as mentioned in *Results*, both representation approaches could be combined in future work. It may happen that each classifier (using different data-representation) performs better in a certain region of the feature space.

Regarding the classification methods, there are some methods that should not be used at all, such as KNN, for its low accuracy and high computational demand. For future work we also consider the inclusion of other methods such as Supported Vector Machine or Random Forest.

The decision tree highlights the relevance of the four following features: night stay, stay duration, start time, and battery status. Therefore we could also classify almost with the same accuracy with fewer features, which is more efficient in terms of time and computation.

An important design constraint was the requirement is that the features are accessible by phone vendors and the features can be used to solve real problems without violating the user's privacy. In a more general sense, this method is not meant to be used independently for context detection. One could combine the method presented in this work with any other methods that combine information from social network (e.g., Facebook status) or the usage of mobile applications (e.g., using Bing Maps provides certain information about user's context). However, the access to this complementary information requires complying with user's privacy requirements.

ACKNOWLEDGEMENT

This work was financially supported by EU FP7 Marie Curie Initial Training Network MULTI-POS (Multi-technology Positioning Professionals) under grant nr. 31652, Nokia Corporation, and Microsoft Corporation. In particular, we thank the Microsoft Devices Positioning team for the guidance and collaboration during the research. The research in this paper used the MDC Database made available by Idiap Research Institute, Switzerland and owned by Nokia.

REFERENCES

[1] Heggestuen, B. 2013. Smarthphone and tablet penetration – Business Insider. Retrieved June 6, 2014: http://www.businessinsider.com /smartphone-and-tablet-penetration-2013-10

[2] B. Rao and L. Minakakis, “Evolution of Mobile Location-based Services,” *Commun. ACM*, vol. 46, no. 12, pp. 61–65, Dec. 2003.

[3] A. Bleicher. Wearable Computers Will Transform Language. IEEE Spectrum, pp. 62. Jun 2014.

[4] N. Kiukkonen, J. Blom, O. Dousse, D. Gatica-Perez, and J. Laurila, “Towards rich mobile phone datasets: Lausanne datacollection campaign,” 2010.

[5] T. M. T. Do and D. Gatica-Perez, “The Places of Our Lives: Visiting Patterns and Automatic Labeling from Longitudinal Smartphone Data,” *IEEE Transactions on Mobile Computing*, vol. 13, no. 3, pp. 638–648, Mar. 2014.

[6] G. D. Abowd, A. K. Dey, P. J. Brown, N. Davies, M. Smith, and P. Steggles, “Towards a Better Understanding of Context and Context-Awareness,” in *Proceedings of the 1st International Symposium on Handheld and Ubiquitous Computing*, London, UK, UK, pp. 304–307. 1999

[7] M. Baldauf, S. Dustdar, and F. Rosenberg, “A Survey on Context Aware Systems,” *Int. J. Ad Hoc Ubiquitous Comput.*, vol. 2, no. 4, pp. 263–277, Jun. 2007.

[8] O. A. Nykänen and A. Rivero-Rodriguez, “Problems in Context-Aware Semantic Computing,” *International Journal of Interactive Mobile Technologies (iJIM)*, vol. 8, no. 3, pp. pp. 32–39, Jun. 2014.

[9] J, Kantola, M. Perttunen, T. Leppänen, J. Collin, and J. Riekki. ”Context awareness for GPS-enabled phones”.*Proceedings of ION ITM* (San Diego, CA, USA)pp.117–124. Jan 2010

[10] L. Pei, R. Chen, J. Liu, W. Chen, H. Kuusniemi, T. Tenhunen, et al. ”Motion Recognition Assisted Indoor Wireless Navigation on a Mobile Phone”. *Proceedings of ION GNSS 2010* (Portland, OR, USA), pp. 3366-3375. Sept 2010.

[11] P. Zhou, Y. Zheng, Z. Li, M. Li, and G. Shen, “IODetector: A Generic Service for Indoor Outdoor Detection,” in *Proceedings of the 10th ACM Conference on Embedded Network Sensor Systems*, New York, NY, USA, 2012, pp. 113–126.

[12] A. Eronen, J. Leppänen, J. T. Collin, J. Parviainen, and J. Bojja. Method and apparatus for determining environmental context utilizing features obtained by multiple radio receivers. U.S. Patent 20 130 053 069

[13] T.-M.-T. Do and D. Gatica-Perez, “By Their Apps You Shall Understand Them: Mining Large-scale Patterns of Mobile Phone Usage,” in *Proceedings of the 9th International Conference on Mobile and Ubiquitous Multimedia*, New York, NY, USA, pp. 27:1–27:10. 2010

[14] Y. Zhu, E. Zhong, B. Wu. “Feature engineering for place category classification,” In *Mobile Data Challenge by Nokia Workshop, in Conjunction with International Conference on Pervasive Computing* (Newcastle, UK) June, 2012.

[15] C.-M. Huang, J.J.-C. Ying, and V. S. Tseng. “Mining Users’ Behaviors and Environments for Semantic Place Prediction” In *Mobile Data Challenge by Nokia Workshop, in Conjunction with International Conference on Pervasive Computing* (Newcastle, UK) Jun 2012.

[16] R. Montoliú Colás, A. Martínez Usó, and J. Martínez Sotoca, “Semantic place prediction by combining smart binary classifiers,” In *Mobile Data Challenge by Nokia Workshop, in Conjunction with International Conference on Pervasive Computing* (Newcastle, UK) Jun 2012.

[17] J. Laurila, “The mobile data challenge: Big data for mobile computing research,” In *Mobile Data Challenge by Nokia Workshop, in Conjunction with International Conference on Pervasive Computing* (Newcastle, UK) Jun 2012.

[18] D. Michie, D. J. Spiegelhalter, and C.C. Taylor. *Machine Learning, Neural and Statistical Classification*. Ellis Horwood Publisher. 1994.