Semantic place prediction by combining smart binary classifiers*

Raul Montoliu
Institute of New Imaging Technologies (INIT)
Jaume I University, Spain
montoliu@uji.es

Adolfo Martínez-Uso
Institute of New Imaging Technologies (INIT)
Jaume I University, Spain
auso@uji.es

Jose Martínez-Sotoca
Institute of New Imaging Technologies (INIT)
Jaume I University, Spain
sotoca@uji.es

ABSTRACT
In this paper, a novel methodology is proposed to predict the semantic meaning of a set of places extracted from location data. A selection of relevant feature families is proposed on the basis of the information collected from users' mobiles phone, whereas the multiclass classificacion problem is addressed by a set of smart binary classifiers. Three different evaluation rules are used: Weighted Voting-based, Error-Correcting Output Codes-based, and the novel Multicoded Class-based. Experiments show the good performance of the methods proposed and the interesting properties of the novel Multicoded class-based evaluation rule.

1. INTRODUCTION
In the last years, the problem of obtaining the coordinates (latitude and longitude) of the places where people go or spend time has been attracted the attention of many researchers [5, 7, 9, 10, 12]. But for many applications, obtaining those coordinates is not enough and semantic information is somehow needed, for instance, the information related to the place category (restaurant, home, work, place to play indoor sports, etc.).

In this paper, a novel methodology is proposed to predict the semantic meaning of a set of places extracted from location data provided by the *Nokia Mobile Data Challenge* (NMDC) for the *Semantic place prediction* task [8]. Table 1 shows the 10 labels of the problem and their semantic meaning. Input data include: the sequence of visits to places where a set of 79 users have stayed for a while (approximately there are 5 labeled places per user), and the information collected from the users' mobile phones [8]. Input data do not include geographic coordinates of the places to protect users' privacy. Thus, given a training data set that includes labeled data, the objective of this work is to obtain an as accurate as possible classification technique that would be able to obtain the semantic meaning of the places visited from another set of users (test set). Training data provide both labeled places and unlabeled places. In this paper, only labeled places have been used.

*This work was supported by the Spanish Ministerio de Ciencia e Innovación under project Consolider Ingenio 2010 CSD2007-00018. This material was prepared for the Mobile Data Challenge 2012 (by Nokia) Workshop; June 18-19, 2012; Newcastle, UK.

Table 1: Labels, semantic meaning and number of labeled prototypes in each class.

Label	Semantic meaning	No. prototypes
1	Home	84
2	Home of a friend	46
3	Work	102
4	Transport related	23
5	Work of a friend	9
6	Indoor sports	25
7	Outdoor sports	14
8	Bar, Restaurant	11
9	Shop	17
10	Holidays resort	5

In our approach, we propose a set of feature families (Section 2) to characterize a place based on the information collected from users' mobile phones. In addition, the multiclass classification problem has been divided into a set of 2-class classification problems, producing an ensemble of smart binary classifiers. The term *smart* refers that, for each binary classification problem, the best combination of features and the best classifier type (between k-Nearest Neighbor (kNN) [1] and Support Vector Machine (SVM) [2]) has been used to train each smart binary classifier. Three different evaluation techniques have been proposed to infer the label of each test sample from the results obtained by all the binary classifiers: Weighted Voting-based (WV), Error-Correcting Output Codes-based (ECOC) [3, 4], and the novel Multicoded class-based (McC).

Summarizing, the main contributions of this paper are as follows:

1. A novel set of features for a semantic characterization of places is proposed. It is based on the information collected from users' mobile phones.
2. A novel methodology based on the use of smart bi-

Table 2: Name of the families and number of features for each one.

Family Name	Based on	No. features
Γ_{time}	Time	288
Γ_{bt}	Bluetooth	288
Γ_{bt+}	Bluetooth	2
$\Gamma_{profile}$	Profile	7
$\Gamma_{battery}$	Battery	2
Γ_{av}	Accelerometer	288
Γ_{av+}	Accelerometer	2
Γ_{vel}	Accelerometer	288
Γ_{vel+}	Accelerometer	2
Γ_{call}	Call log	6
Γ_{sms}	Sms log	2
Γ_{wlan}	Wlan	2

nary classifiers is also proposed to solve the multiclass classification problem.

3. We introduce the novel Multicoded Class-based (McC) multiclass evaluation rule in combination with the also proposed smart binary classifiers methodology. Our proposals outperform the traditional multiclass classification techniques as well as other well-known alternatives of solving the multiclass problem as ECOCs or Weighted voting.

2. THE SET OF FEATURE FAMILIES

In this work, information collected from users' mobile phones has been divided into several *families* in order to characterize the different place categories. Each *family* is understood as a vector of several *features* obtained from the same source. Let us name F to the set of 12 families created to characterize each place. Table 2 shows the name of the families and the number of features included in each one.

It is important to note that only those data detected or collected when the user stayed in the place are used to characterize that particular place.

2.1 Time-based features

People tend to stay or visit places at specific times of the day/week. For instance, it is logical to suppose that users should be at home at night, do not work during the weekends or go to the restaurant at lunch time. Therefore, time-based features can be important to discriminate among classes.

For each place, the input data provide the time periods when the user has stayed in a particular place [8]. Using this information, the probability that the user has stayed in a particular place at a particular time moment has been estimated. To this end, the day has been split into 144 time periods of 10 minutes each. In addition, we estimated separately the probability of staying in places during the working days in contrast to the weekends. The final time-based feature, call Γ_{time}, has 288 elements, first 144 from working days and last 144 from weekends. Families Γ_{bt}, Γ_{av} and Γ_{vel} use the same time periods than Γ_{time}.

2.2 Bluetooth-based features

From raw bluetooth data, two families of features have been estimated, Γ_{bt} and Γ_{bt+}. The first one, is a vector of 288 features where each one is the average of the number of bluetooth devices detected by the user's phone in a particular time period. The second one, is a vector of 2 features: the average of the number of bluetooth devices detected by the user during a complete working day and the same estimation for the weekends.

2.3 Profile-based features

Phone operating system provides 7 different user profiles. The family $\Gamma_{profile}$ encodes the percentage of time that each profile was selected when the user stayed in a place.

2.4 Battery-based features

$\Gamma_{battery}$ family has two features that encode the percentage of time that the phone has been charging the battery and the percentage of time that it has not been charging.

2.5 Accelerometer-based features

Four families have been derived from accelerometer raw data. Γ_{av} and Γ_{vel} of a particular time period are the average *avdelta*[1] and the average velocity respectively. Γ_{av+} and Γ_{vel+} resume the data using only two features, the first one averages Γ_{av} and Γ_{vel} features for the working days whereas the second one does the same for the weekends.

2.6 Call log-based features

Γ_{call} has 6 features: average number of incoming calls, average number of incoming calls to numbers included in the address book, average duration of incoming calls, and the same three for outcoming calls.

2.7 Sms log-based features

Γ_{sms} has two features: average number of incoming short messages (sms) and the same for outcoming ones.

2.8 Wlan-based features

Γ_{wlan} has two features: the average number of wlan devices and the average number of open wlan devices detected by the phone.

3. THE PROPOSED METHOD

One of the traditional ways of solving a multiclass classification problem is to use a multiclass classifier. An alternative way consists on dividing the problem into a set of binary classification problems. Let us define $\phi_{p,n} = \{\delta_1, \ldots, \delta_D\}$ as the set of all possible binary problems (D) that can be formulated within a 10-class classification problem, where p and n are the number of positive and negative classes, respectively, involved in each binary problem. For instance, $\phi_{1,1}$ (also known as *one-versus-one*) and $\phi_{1,2}$ are composed by the following binary problems:

$$\begin{aligned}
\phi_{1,1} &= \{[\{1\},\{2\}],[\{1\},\{3\}],\ldots,[\{9\},\{10\}]\} \\
\phi_{1,2} &= \{[\{1\},\{2,3\}],[\{1\},\{2,4\}],\ldots,[\{8\},\{9,10\}]\}
\end{aligned}$$

[1] *avdelta* measure is the given value that summarizes the activity of the accelerometer during a time period, see [8] for details of how it is worked out.

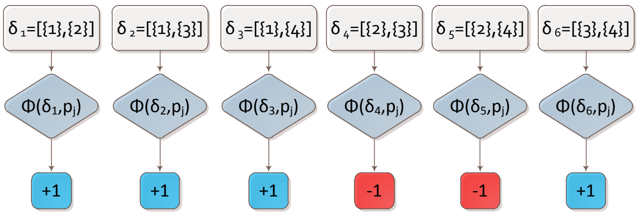

Figure 1: Example of the proposed classification scheme using a set of *smart* binary classifiers. In this case, there are 4 classes, and the set $\phi_{1,1}$ is used ($D = 6$). For a test sample p_j the output vector is obtained (for instance) as $H_j = \{+1, +1, +1, -1, -1, +1\}$.

The expression $\delta_i = [\{a_1^i, \dots, a_p^i\}, \{b_1^i, \dots, b_n^i\}]$, where $i \in [1, \dots, D]$, indicates the positive (labeled as $+1$) and negative (labeled as -1) classes that define a binary problem, being $\{a_1^i, \dots, a_p^i\}$ and $\{b_1^i, \dots, b_n^i\}$ the set of p positive and n negative classes, respectively, involved in i-th binary problem. It is worth noticing that those classes that are used together are merged as if they were a single class. Also note that the sets $\phi_{p,n}$ and $\phi_{n,p}$ are equivalent.

For each binary problem δ_i, an evaluation score have been estimated for all combinations of the families included in the set F and for each binary classifier type. Each family has its own feature vector and, concatenating two or more of these vectors, new features are created that can be used for classifying the different places. For instance, combining Γ_{time} and Γ_{bt} features, a new feature vector with 576 elements is created (i.e. 288 from Γ_{time} and 288 from Γ_{bt}).

The objective is to select the set of features and the binary classifier type that achieve the best performance for each binary classifier. A 2-fold evaluation strategy, has been used in this step.

Each binary problem δ_i is evaluated by the score $\alpha_i = \frac{TP(i)+TN(i)}{2}$, where $TP(i)$ and $TN(i)$ are respectively the true positive rate and the true negative rate of the δ_i binary problem. Accuracy has not been used as score since some binary problems (e.g. [{3}, {10}]) are strongly unbalanced. We look for binary classifiers able to correctly classify samples belonging to both parts of the binary problem. The final classifier is an ensemble of D *smart* binary classifiers.

Test samples are now evaluated to obtain the desired label in the set $[1, \dots, 10]$. For each test sample ρ_j, (where $j \in [1, \dots, Q]$ and Q is the number of test samples), a vector H_j of D labels in the set $[+1, -1]$ are obtained using $H_j(i) = \Phi(\delta_i, \rho_j), \forall i$, i.e. evaluating the test sample using the decision function $\Phi(\delta_i, \rho_j)$ of each binary problem δ_i. Figure 1 shows an illustrative example of the proposed scheme for a hypothetical classification problem with 4 classes.

Three evaluation rules have been proposed to obtain the predicted label from vector H_j: Weighted Voting-based (WV), Error-Correcting Output Codes-based (ECOC) and Multi-coded class-based (McC). They are explained in the following sections.

3.1 Weighted Voting-based (WV) rule

Let us define V^+ and V^- ($V^+, V^- \in \mathbb{N}^{10}$, since 10 are the number of classes in our problem) as two vectors where the votes of each binary classifier are going to be accumulated. If $H_j(i)$ is positive, then a positive vote is accumulated for all classes belonging to $\{a_1^i, \dots, a_p^i\}$. At the same time, a negative vote is accumulated for all classes belonging to $\{b_1^i, \dots, b_n^i\}$. If $H_j(i)$ is negative, the procedure is the opposite. The next step is to look for the maximum in $(V^+ - V^-)$. The most probable class for the test sample is the one related with the position giving the maximum in $(V^+ - V^-)$. The amount of the votes are determined by $TP(i)$ and $TN(i)$, obtained for each binary classification problem.

According to the example shown at Figure 1 and assuming for simplicity that $TP(i) = 1$ and $TN(i) = 1$, $\forall i$ (i.e. all binary classifiers having the same weight), then $V^+ = (3, 0, 2, 1)$ and $V^- = (0, 3, 1, 2)$ ($V^+, V^- \in \mathbb{N}^4$, since a 4-classes classification problem has been used in this example for simplicity). For instance, $V^+(1) = 3$ since 3 binary classifiers (δ_1, δ_2 and δ_3) score $+1$ when the class 1 belongs to the positive part of the binary problem (i.e to $\{a_1^i, \dots, a_p^i\}$). $V^-(1) = 0$ since there are not binary classifiers scoring -1 when the class 1 belongs to the negative part of the binary problem (i.e. to $\{b_1^i, \dots, b_n^i\}$). Similarly, $V^+(3) = 2$ since 1 binary classifier (δ_6) scores $+1$ when the class 3 belongs to the positive part of the binary problem and 1 binary classifier (δ_4) scores -1 when the class 3 belongs to the negative part. $V^-(3) = 1$ since 1 binary classifier (δ_2) scores -1 when the class 3 belongs to the negative part of the binary problem.

The label estimated for the test sample p_j is determined by the position of the maximum of $(V^+ - V^-) = (3, -3, 1, -1)$, i.e. in this example, the label of the first class is the one that will be selected for the $j - th$ test sample.

3.2 Error-Correcting Output Codes-based (ECOC) rule

ECOC techniques [3, 4] were designed as an alternative way of combining binary problems in order to deal with the multi-class case. Given a set on N classes to be learnt ($N = 10$ in our problem), in a binary symbol-based ECOC framework, D different bi-partitions (groups of classes) are formed, and D binary problems over the partitions are trained. As a result, a codeword of length D is obtained for each class, where each position of the code corresponds to a response of a given dichotomizer (coded by $+1$ or -1 according to their class set membership).

During the decoding process, applying the D binary classifiers, a code x (H_j in our terminology) is obtained for each data sample in the test set. This code x is compared to the codewords of each class, being assigned to the class with the closest codeword. In the ternary symbol-based ECOC, the code symbols are in the set $\{+1, -1, 0\}$. In this case, the symbol zero means that a particular class is not considered for a given classifier.

In our work, a ternary symbol-based ECOC has been used, using the set of binary problems $\phi_{p,n}$ as dichotomizers. Figure 2 shows an illustrative example of a ternary symbol-

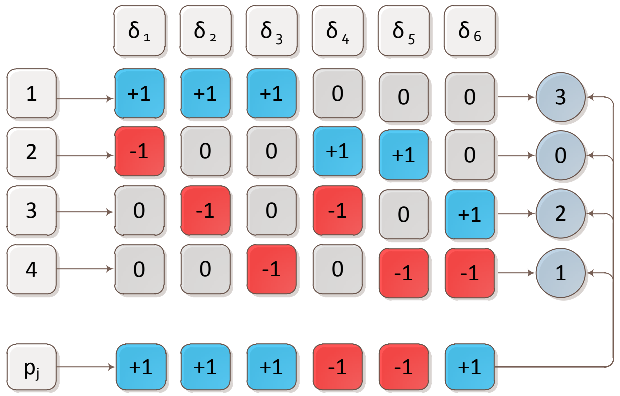

Figure 2: Example of a ternary symbol-based ECOC for a classification problem with 4 classes. Rows show the codification of each class and columns the $D = 6$ dichotimizers. Each position of the code is coded by $+1$ (blue), -1 (red) or 0 (gray), according to their class set membership. Last row shows the decodification (H_j) for the test sample p_j. Circles show the resulting distance between each code and the test sample. In this example, the number of equal symbols has been used as distance. The label of the first class is the one that will be selected for the $j-th$ test sample.

based ECOC for a hypothetical classification problem with 4 classes.

3.3 Multicoded Class-based (McC) rule

In the original ECOC method, a code is obtained for each data sample in the test set applying the D binary classifiers. We propose a new evaluation rule which consists on also assigning a code to the training samples by applying the D binary classifiers. Thus, both the test samples and the training samples will be expressed in the *code* space, transforming the set of smart binary classifiers into a multiclass problem with the original 10 classes. The next step is to apply a multiclass classifier to obtain the labels in the set $[1, \ldots, 10]$.

4. EXPERIMENTS AND RESULTS

4.1 Experiments set-up

The input database provided by the NMDC consist on a set of approximately 5 labeled places by user. In particular there are 336 labeled places. The number of places is not 79x5=395 since some users have less than 5 labeled places. A 2-fold cross validation strategy has been selected to evaluate the different classifiers that will be explain along this section. Therefore the input data has been divided in two sets, the first one *setA* with 171 samples and the second one *setB* with the remaining 165. The prototypes included in each set have carefully been chosen to maintain a similar number of prototypes of each class in each set.

KNN and SVM multiclass and 2-class classifiers are used along this work. On the one hand, using the well-known k-nearest neighbor method [1], each new sample is classified by calculating the distance to the k-nearest training samples. The class label of the new sample is determined by using a majority-voting scheme. On the other hand, SVM [2] uses a kernel to transform the original data in a transformed space when a linear classifier is applied.

In this work the LIBSVM package [6] has been used, which supports both 2-class and multiclass classification. The kernel used has been the Radial Basis Function (RBF) both for 2-class and multiclass classification. In SVM, it is not known beforehand which parameters are best for a given problem, consequenlty an estimation of the parameters must be done. For this purpose, a "grid-search" on parameters C and σ has been performed using cross validation. Various pairs of C and σ values (using exponentially growing) have been tried and the one with the best score has been picked. In addition, *prtools* software[2] [11] has been used both for Knn and for the rest of classifiers used in this work.

4.2 Baseline

In order to obtain a baseline classification, all the possible combinations of the families from set F, that is 4095 combinations, are tested in order to find the best space for discriminating among them. For each combination, a 2-fold evaluation procedure is performed to obtain the accuracy (number of well classified test samples divided by the total number of test samples) of the classifier. The best combination is the one with higher accuracy. The same procedure has been performed using different types of classifiers.

Table 3 shows the results obtained in terms of accuracy for the baseline multiclass classification problem, using several multiclass classifier types. It shows only the accuracy of the best combination of features for each classifier type. SVM multiclass classifier is the one that obtains better accuracy. The feature space in this case, has been obtained by concatenating the families Γ_{time}, Γ_{bt}, Γ_{bt+}, $\Gamma_{profile}$, Γ_{vel} and Γ_{wlan}, producing a vector of 875 elements.

4.3 Proposed methods

Table 4 shows the results obtained in terms of accuracy for the proposed method applying the three evaluation rules. Three sets of binary problems have been tested: $\phi_{1,1}$ ($D = 45$), $\phi_{1,2}$ ($D = 360$) and $\phi_{1,1} \cup \phi_{1,2}$ ($D = 405$). Each smart binary classifier has been evaluated using the 1NN and the SVM 2-class classifiers.

In the ECOC-based rule, a Manhattan distance has been used to compare the code of a test sample with the codewords of each class.

In McC rule, a SVM multiclass classifier has been used to obtain the labels in the *code* space.

In addition, a new method that consists on combining the results of the baseline technique with the three results obtained using the proposed method (one for each evaluation rule) is also proposed. In this ensemble, the class label of test samples are determined by using a majority-voting scheme.

4.4 Discussion

[2]http://www.prtools.org

Table 3: Accuracy of the multiclass classification problem (baseline).

Classifier	1NN	SVM	Parzen	Binary decision tree
Accuracy	59.86	**65.82**	61.67	54.10

Table 4: Accuracy of the proposed method and the final ensemble

	$\phi_{1,1}$	$\phi_{1,2}$	$\phi_{1,1} \cup \phi_{1,2}$
VW-based rule	66.70	71.16	70.88
ECOC-based rule	66.70	70.56	70.57
McC-based rule	67.35	71.19	71.17
Final ensemble	67.31	72.96	**73.26**

From the results shown at Tables 3 and 4 we would like to stress the following ideas:

- To divide the multiclass problem into a set of smart binary problems improves the accuracy of the classification. The three evaluation rules obtain quite similar results, being the McC-based rule the best one.
- The use of the set of binary problems $\phi_{1,2}$ instead of the traditional $\phi_{1,1}$ improves the accuracy for all methods. However, when the union of both sets has been used, the accuracy does not always outperform the case when only the set $\phi_{1,2}$ has been used. This means that the use of the $\phi_{1,1}$ does not always introduce new valuable information to improve the accuracy.
- The final ensemble, which is a combination of the predictions of the previous methods, outperforms the results obtained when the sets $\phi_{1,2}$ and $\phi_{1,1} \cup \phi_{1,2}$ are used, **obtaining the best result in this work**, with an accuracy of 73.26%.
- Figure 3 shows the confusion matrix obtained by using the final ensemble method and the set $\phi_{1,1} \cup \phi_{1,2}$. In general, the method provides excellent performance on *Home* (label 1) and *Work* (label 3) classes. It is worth noticing that *Home of a friend* (label 2) and *Work of a friend* (label 5) classes are sometimes confused with the *Home* and *Work* classes, respectively. The method has low accuracy for the classes *Outdoor sports* (label 7) and *Bar, Restaurant* (label 8), while obtaining good performance recognizing prototypes from the classes *Transport related* (label 4), *Indoor sports* (label 6) and *Holidays resort* (label 10).
- If we look the labels predicted by the baseline and the 3 evaluation rules of our proposed method (not shown in this paper), we realize that the labels predicted by the baseline method and the McC-based rule are quite similar (but being the accuracy when using the McC-based rule significantly better) and the labels predicted by VW-based and ECOC-based rules are also quite similar. Note that the baseline method and the McC-based rule solve the multiclass problems by using a multiclass classifier (SVM in both cases) where VW-based and ECOC-based rules solve the multiclass problem by dividing the problem into a set of binary problems. Then, we can conclude that the proposed McC-based is an interesting alternative to directly solve the multiclass problem that produces good accuracy results on contrast to other traditional ways of solving the multiclass classification problem.

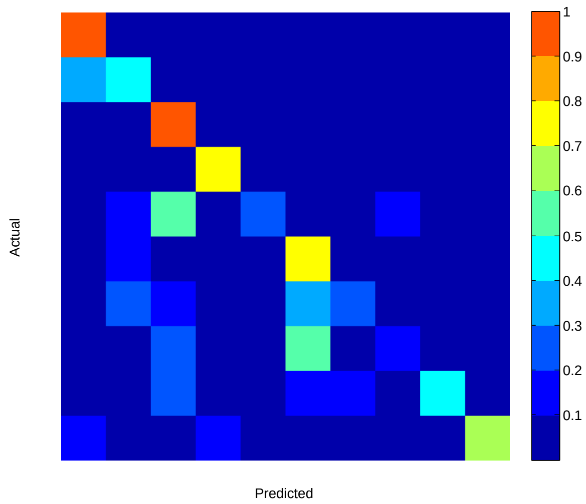

Figure 3: Confusion matrix obtained by using the final ensemble method and the set $\phi_{1,1} \cup \phi_{1,2}$. For a better visualization of this figure, It is recommended to read the color version of the article.

- In ECOC-based rule all the training samples from the same class are represented using the same codeword, i.e. they fall in the same place on the decision frontier of the classifier. In contrast, in the proposed McC-based rule, each class can be represented using different codewords, therefore it improves how each class is represented in the feature space, giving a better versatility on obtaining the label of the test samples.
- In the proposed method, when the VW-based and ECOC-based evaluation rules have been used, the multiclass classification problem is transformed by dividing the problem into a set of smart binary classification problems, therefore each individual binary problem has a decision frontier easier to be estimated in contract to the multiclass problem. However, this simplification implies that the uncertainty is transferred to the WV-based or to the ECOC-based systems. McC-based rule solves this situation acting as a multiclass classifier significantly improving traditional multiclass classifiers (baseline) and slightly improving to two popular alternatives to solve the multiclass classification problem as the VW-based and ECOC-based evaluation strategies.

Figure 4 shows the most important feature families, i.e. the ones that have been most frequently selected in by the smart binary classifiers. In general, Γ_{time}, Γ_{wlan}, Γ_{call}, $\Gamma_{profile}$ and Γ_{sms} are the most important ones. On the other hand, the 1NN classifier has been selected as the binary classifier at the 81% of the times, where the SVM one, only at 19%.

5. CONCLUSIONS

In this paper, a novel methodology has been proposed to predict the semantic meaning of a set of places extracted

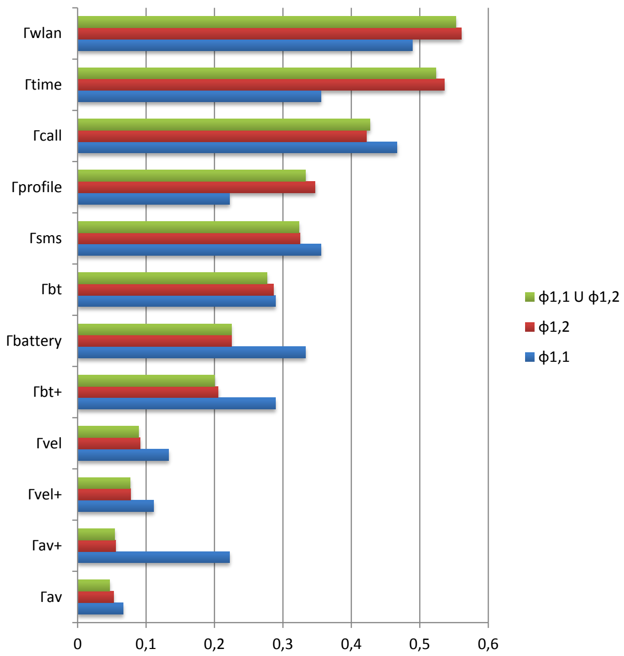

Figure 4: Percentage of times that the feature families have been selected to be part of the feature vector of the smart binary classifiers included in the sets $\phi_{1,1} \cup \phi_{1,2}$ (first/green bar), $\phi_{1,2}$ (second/red bar) and $\phi_{1,1}$ (third/blue bar).

from location data provided by the *Nokia Mobile Data Challenge* for the *Semantic place prediction* task [8]. A novel set of feature families have been proposed on the basis of the information collected from users' mobile phones. Different classification strategies to deal with the multiclass problem have been tested. Experiments have shown the good performance of the method proposed and the interesting properties of the novel Multicoded Class-based evaluation rule. Future work will focus on obtaining additional features from raw data, on improving decision frontiers of the binary problems and on studying with more detail the interesting properties arisen from the experiments performed with the novel McC-based classification technique.

6. REFERENCES

[1] T. Cover and P. Hart. Nearest neighbor pattern classification. *IEEE Trans. on Information Theory*, 13(1):21–27, 1967.

[2] N. Cristianini and J. Shawe-Taylor. *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*. Cambridge University Press, 1 edition, 2000.

[3] T. Dietterich and G. Bakiri. Solving multiclass learning problems via error-correcting output codes. *J. Artificial Intelligence Research*, 2:263–286, 1995.

[4] S. Escalera, O. Pujol, and P. Radeva. On the decoding process in ternary error-correcting output codes. *IEEE Trans. on PAMI*, 32(1):120–134, 2008.

[5] J. Hightower, S. Consolvo, A. LaMarca, I. Smith, and J. Hughes. Learning and recognizing the places we go. In *Proc. of Ubicomp '05*, pages 159–176, 2005.

[6] C.-W. Hsu and C.-J. Lin. A comparison of methods for multi-class support vector machines. *IEEE Trans. on Neural Networks*, 13:415–425, 2002.

[7] D. H. Kim, J. Hightower, R. Govindan, and D. Estrin. Discovering semantically meaningful places from pervasive rf-beacons. In *Proc. of Ubicomp '09*, pages 21–30, 2009.

[8] J. K. Laurila, D. Gatica-Perez, I. Aad, J. Blom, O. Bornet, T.-M.-T. Do, O. Dousse, J. Eberle, and M. Miettinen. The mobile data challenge: Big data for mobile computing research,. In *Mobile Data Challenge by Nokia Workshop, in conjunction with Int. Conf. on Pervasive Computing*, June 2012.

[9] R. Montoliu, J. Blom, and D. Gatica-Perez. Discovering places of interest in everyday life from smartphone data. *Multimedia Tools and Aplications*, In press.

[10] R. Montoliu and D. Gatica-Perez. Discovering human places of interest from multimodal mobile phone data. In *Proc. of MUM '10*, 2010.

[11] F. van der Heijden, R. Duin, D. de Ridder, and D. M. J. Tax. *Classification, Parameter Estimation and State Estimation: An Engineering Approach Using MATLAB*. Wiley, 1 edition, 2004.

[12] V. W. Zheng, Y. Zheng, X. Xie, and Q. Yang. Collaborative location and activity recommendations with gps history data. In *Proc. of WWW '10*, pages 1029–1038, 2010.