

Mobile Activity Recognition using Ubiquitous Data Stream Mining

João Bártolo Gomes¹, Shonali Krishnaswamy¹, Mohamed M. Gaber²,
Pedro A.C. Sousa³, Ernestina Menasalvas⁴

¹ Institute for Infocomm Research (I2R), A*STAR, Singapore
bartologjp@i2r.a-star.edu.sg
spkrishna@i2r.a-star.edu.sg

² School of Computing, University of Portsmouth, United Kingdom
mohamed.gaber@port.ac.uk

³ Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa, Portugal
pas@fct.unl.pt

⁴ Facultad de Informatica, Universidad Politecnica Madrid, Spain
emenasalvas@fi.upm.es

Abstract. Mobile activity recognition focuses on inferring the current activities of a mobile user by leveraging the rich sensory data that is available on today's smart phones and other wearable sensors. The state of the art in mobile activity recognition research has focused on traditional classification learning techniques. In this paper, we propose the Mobile Activity Recognition System (MARS) where for the first time the classifier is built on-board the mobile device itself through ubiquitous data stream mining in an incremental manner. The advantages of on-board data stream mining for mobile activity recognition are: i) personalisation of models built to individual users; ii) increased privacy as the data is not sent to an external site; iii) adaptation of the model as the user's activity profile changes. In our extensive experimental results using a recent benchmarking activity recognition dataset, we show that MARS can achieve similar accuracy when compared with traditional classifiers for activity recognition, while at the same time being scalable and efficient in terms of the mobile device resources consumption. MARS has been implemented on the Android platform for empirical evaluation.

1 Introduction

The integration of small wireless sensors into objects of everyday life allows to create a non-intrusive sensory data rich environment. Thus, the miniaturisation and cost reduction of sensor hardware and mobile devices has led to the emergence of research into mobile AR [1]. In several existing studies wearable sensors are used by people while performing their daily activities [1, 6], while others additionally use sensors embedded into tools and utensils in an apartment, which allows the analysis of more fine grained activities [8].

Mobile AR is usually formulated as a classification problem, where supervised machine learning is used to interpret sensed data into activities [1, 7]. The learning

process normally goes through the following stages: i) data collection, where sensor data is collected over a specified period of time from one or more mobile users, with the users typically labelling/annotating their activities; ii) data transfer, where the collected data is transferred to and collated in a centralised repository; iii) learning/model building, where the AR classification model is trained and tested using the collected data; iv) model deployment, where the learnt model is deployed on-board the mobile device for identifying and classifying activities from sensory data. These state of the art mobile AR approaches from ubiquitous sensors have been shown to achieve high recognition rates [7]. This may give the impression that the general problem of AR has been solved successfully. However, in existing approaches the obtained models are static, are built off-line in an external (to the mobile device) environment and little attention is given to issues such as personalisation of generic models and privacy.

To address these issues in this paper, we propose the Mobile Activity Recognition System (MARS) that learns the classification model on-board the mobile device itself through ubiquitous data stream mining in an incremental manner. The proposed system (MARS) has been implemented on the Android platform to evaluate its feasibility.

The rest of the paper is organised as follows. The following Section reviews the related work. Section 3 presents the definition of mobile AR as classification problem, which is followed by a detailed description of the existing open challenges in Section 4. The proposed Mobile Activity Recognition System (MARS) is presented Section 5. The experimental setup and results are discussed in Section 6. Finally, in Section 7, conclusions of this work and future work are presented.

2 Related work

AR from sensor data is a popular research field that has contributed with several high recognition rate approaches. Many of these use supervised machine learning algorithms, such as Decision Trees [1], Artificial Neural Networks, Hidden Markov Models, Naive Bayes, K-Nearest Neighbour or Support Vector Machines. For an extensive review of supervised learning approaches for AR please refer to [7]. Here we focus our review on works that perform mobile AR from sensor data.

One of the most cited publications on activity recognition in pervasive computing [1] deployed five small biaxial accelerometers worn simultaneously on different body positions in order to distinguish 20 activities of interest. The data was collected from 20 subjects that annotated it themselves without researcher supervision or observation. From the learning algorithms tested, C4.5 decision trees showed the best performance with an overall accuracy rate of 84%. Such technique is considered to be slow to train but quick to run. Therefore, the authors suggest that a pre-trained decision tree should be able to recognise user activities in real-time on a 2004 top-end mobile device. Moreover, it is reported that some activities are recognised with subject-independent training data while others seem to require subject-specific training data.

In a recent paper, [6] proposes and experimentally evaluates a system that uses phone-based accelerometers to perform mobile AR. Data was collected from 29 subjects as they performed their daily activities such as walking, jogging, climbing stairs, sitting, and standing. This work shows how a smart phone (Android) can be used to perform activity recognition, simply by carrying it in a fixed position (front pants leg pocket). The results show that most activities are recognised correctly over 90% of the time. Still, the collected data is transferred to an Internet-based server where a static model is generated off-line. Again the issues of personalisation or privacy are not addressed but in the future work section it is mentioned that an improvement of the proposed system would be to generate the model on-board. Nevertheless, to the best of our knowledge such improvement has not yet been proposed.

The reviewed approaches built static classification models off-line in an external (to the mobile device) environment. Moreover, the streaming nature of data is not taken into account nor the possibility that the model needs to be adapted over time. In addition, little attention has been given to the personalisation of the built model to suit a particular user, despite the results that seem to indicate that better accuracy is obtained with personalised models (i.e., training and test data from the same subject). To the best of our knowledge, no other ubiquitous data stream mining approach has been proposed so far to address mobile AR.

For a more extensive review of data stream mining systems that have been used successfully in other applications please refer to [5]. A demo of MARS has been recently presented in [2].

3 Problem definition

Let X be the space of features that correspond to the available input sensor features and Y be the set of possible (discrete) class labels that correspond to the activities of interest. Consider a data stream DS , where $X_i = (\mathbf{x}_i, y_i)$ with $x_i \in X$ and $y_i \in Y$, represents the i^{th} record in DS . The modelling of AR is formulated as a function f that assigns each sensor feature input record \mathbf{x}_i to the true activity label y_i . This function f can be approximated using supervised learning by training a model m . The goal is that the trained model m minimises the number of wrongly recognised activities (i.e., achieves high accuracy).

4 Open challenges

Despite the good results of existing supervised learning approaches in AR, there are still open challenges that to the best of our knowledge have not been addressed. The following subsections introduce such challenges.

4.1 Training data

The usual supervised learning approach to AR assumes that there is abundant training data and that the function f to model is static. However, in realistic

situations, f is usually subject dependent and can even change over time within subject. Moreover, past work shows that if the training data is collected from the subject of interest then there is no advantage to use additional training data from other subjects [1, 7, 6]. Still, in the case where training data from the subject of interest is not available, having data from higher number of subjects is beneficial to the resulting recognition accuracy.

4.2 Model generation

In most existing supervised learning approaches to mobile AR, the training data is collected, a classification model is generated offline from the collected data, and finally the obtained model is deployed. Nevertheless, there are disadvantages that can result from using this type of offline learning process:

- The obtained model is static - Once a model is generated it cannot incorporate new information.
- Computational costs - The batch algorithms typically used to generate the model are not designed to be executed in mobile devices. Such algorithms usually require several passes over the dataset and require that the entire dataset is allocated into main memory. In contrast, ubiquitous data stream mining approaches process each record only once and are memory efficient [5].
- Accuracy assessment in a realistic scenario - The static model that is deployed can have good accuracy on a testing set which is usually similar to the training set, when methods such as cross validation or a hold-out set are used. However, the performance of this model in a realistic situation depends on how the test set is representative of the usage scenario.

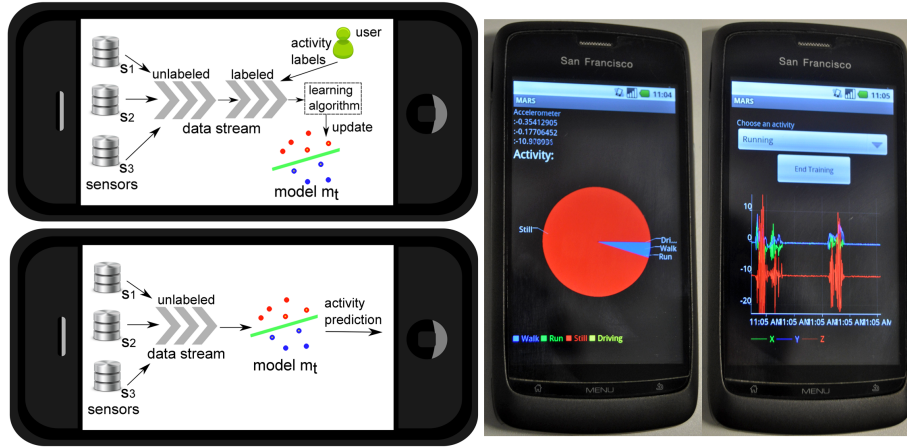


Fig. 1: MARS: framework and implementation

5 MARS: Mobile Activity Recognition System

This paper proposes MARS, a ubiquitous data stream mining approach to mobile AR. Such approach is motivated and focused on addressing the open challenges described in the previous section. Conversely to traditional supervised learning, data stream classification algorithms are able to update an anytime model m_t as new training records are available in the stream. Moreover, these algorithms are light-weight and can be executed using the computational resources usually available on nowadays mobile devices. The proposed approach enables greater personalisation and privacy while bringing the whole learning process on-board the mobile device.

The learning process is divided into two phases:

- Training - During the training phase the user performs the activities of interest, either in predefined drills or freely during normal activities and annotates interactively the data collected from the sensors using a user-friendly interface (i.e., usually simply by selecting from a list the activity that he previously executed). This type of naturalistic data collection has been successfully used before, however, the records are saved to be then processed by a offline learning algorithm, while we propose that the annotated data stream should be processed on-board by an incremental learning algorithm. Moreover, since an anytime model m_t (i.e., model at time t) is assumed, it is possible to estimate the accuracy of this model as new records are incorporated. For this purpose we propose that the prequential statistic is used [4]. Figure 1 illustrates MARS training phase, where the data (unlabelled) is coming from the available sensors, then the user annotates/labels such data which is processed by the learning algorithm that updates the anytime model m_t .
- Activity Recognition - the new records (unlabelled) to be classified are given to the anytime model m_t that returns the predicted activity. This phase is also illustrated in Figure 1.

6 Experiments

This section describes the experiments that were performed to evaluate the MARS feasibility and accuracy. The data used in the experiments has been released for the OPPORTUNITY AR challenge¹, which aims to provide a comparative benchmark dataset for AR approaches.

6.1 The AR challenge data

The lack of established benchmarking problems for AR is one of the motivations behind the OPPORTUNITY AR challenge [8]. The data contains daily human activities recorded in a sensor rich environment: a room simulating a studio flat with kitchen, deckchair, and outdoor access where subjects performed daily

¹ <http://www.opportunity-project.eu/challenge>

morning activities. Two types of recording sessions were performed: Drill sessions where the subject performs sequentially a pre-defined set of activities and "daily living activities" runs (ADL) where the subject executes a high level task (wake up, groom, prepare breakfast, clean) with more freedom about the sequence of individual atomic activities. It records 72 sensors of 10 modalities, integrated in the environment, in objects, and on the body. It consists of an annotated dataset of complex, interleaved and hierarchical naturalistic activities, with a particularly large number of atomic activities (around 30.000), collected in a rich sensor environment. Data was manually labelled during the recording and later reviewed by at least two different persons based on the video recording.

The data used for the challenge is composed of the recordings for 4 subjects. For each subject there are 5 unsegmented sessions. During the challenge, for 3 of the 4 subjects the last two sessions were used by the organisers to evaluate the performance of the contributed methods. Moreover, the subject number 4 is used to assess robustness to noise, as rotational and additive noise has been added to the test data (last two sessions) of this subject.

The challenges consisted of 4 challenges, but for the purpose of this work evaluation we will consider the multimodal classification tasks that are:

- Modes of locomotion (Task A) - The goal of this task is to classify the subject mode of locomotion (i.e., stand, walk, sit, lie) from body-worn sensors.
- Gestures (Task B2) - This task concerns recognition of right-arm gestures (17 classes) performed in a daily activities scenario. Gestures include, clean the table, open/close a door/fridge/dishwasher/drawer and toggle a switch.

For the experiments we followed the evaluation proposed in the challenge, that is, the last two sessions were used for testing and the Drill session plus the first 3 ADL sessions were used for training. Note that we are also using data from subject 4 that contains noise in the test set (last two sessions). Therefore, we can also assess how the approach deals with noise, which is somehow similar to the challenge Task C which used data from subject 4, instead of subjects 2 and 3, to assess the accuracy for the Gestures classification task.

6.2 Implementation

To evaluate the feasibility of the MARS, we implemented a prototype on the Android platform. The experiments were carried out on a low-end Android phone, ZTE Blade, sold in UK as Orange San Francisco, that in early 2011 was one of the budget Android phones on the market. The phone has a Qualcomm MSM7227 600 MHz processor, 512MB of RAM, 1250 mAh battery and runs Android 2.2 Froyo.

The learning algorithms used in the experiments (i.e., the incremental Naive Bayes and C4.5 decision tree) are available in the WEKA. Since these algorithms are developed in Java it was easy to port them to the Android application. However, during our preliminary tests we noted it was not even possible to execute the decision tree algorithm (implemented in WEKA as J48) without

getting a memory exception (`java.lang.OutOfMemoryError`), as this algorithm requires that all the dataset is loaded into memory. The max heap size for an Android application depends on the device but if the application is supposed to run on any device the memory allocation should be kept under 16MB. Since each AR challenge session file ranges from 10MB to 33 MB and the training set for each subject contains 4 sessions. Consequently, we had to perform the accuracy measures and running time experiments for the decision tree algorithm on a laptop computer with a 2.10 GHz Intel Core 2 Duo processor and 4GB of RAM memory.

The incremental Naive Bayes approach executed on the device without problems. The data files are read sequentially as a stream to incrementally train the classification model. Then this model is used to sequentially classify the session files that belong to the test set. During this process the only memory consumed is the one required to keep the anytime model and to read the record to be processed, that is then subsequently freed from memory.

The learning algorithm is executed in a dedicated thread that can run in background as a service and therefore the anytime classification model can be updated or asked for prediction at anytime.

6.3 Incremental vs Traditional

Here we report the experiments where we compared in terms of accuracy and running time the incremental Naive Bayes (NB) approach and the J48 Decision Tree (DT). We decided to compare the proposed approach with DT because this algorithm have been shown high accuracy in AR problems and was the first ranked algorithm on the locomotion predictive task of the OPPORTUNITY AR challenge (task A).

Table 1 summarises the results for the different subjects (rows) on the two classification tasks (i.e., locomotion and gestures) that are shown on the respective column. The first thing noticed is that for both algorithms there is higher accuracy on the locomotion task than in the gestures task. This is justified by the fact that the gestures task is more demanding, that is, has more classes (17 instead of 4) and the available sessions have less annotated data for this task than the locomotion one. Nevertheless, these results are within what has been reported in the literature for state of the art approaches in AR [7] and what has been reported recently for the OPPORTUNITY AR challenge [8].

Looking at the accuracy results between the NB vs DT, we observe that for the locomotion task the mean accuracy for the NB is $86,4\% \pm 3,9$ while it is slightly lower for the DT with $85,9\% \pm 2,5$. For subjects S1 and S2 the incremental NB even achieves higher accuracy than the DT. When the gestures task is considered, the mean accuracy for the NB is $56,1\% \pm 5,4$ and the DT obtains $60,9\% \pm 7,1$. Still, the simple incremental NB approach achieves similar accuracy, particularly for subject S3 where only 1.1% difference exists. Moreover, subject 4 achieves the lowest accuracy on both tasks, due to the presence of noise, which seems to have a minor impact on the accuracy.

	Locomotion		Gestures	
	NB	DT	NB	DT
S1	91.5%	88.8%	61.7%	69.1%
S2	87.4%	86.8%	53.3%	54.4%
S3	84.4%	85.5%	59.4%	64.4%
S4	82.3%	82.7%	50.0%	55.7%

Table 1: Accuracy results of NB vs DT

	Locomotion		Gestures	
	NB	DT	NB	DT
S1	21.91s	664.09s	6.48s	127.28s
S2	18.93s	564.03s	6.11s	151.62s
S3	23.67s	793.60s	6.56s	119.46s
S4	16.78s	546.70s	4.65s	100.82s

Table 2: Training times of NB vs DT

Table 2 summarises the training times for obtaining the models for the different subjects (rows) and tasks/learning algorithms in columns. The training times, show that on both tasks the NB is much faster than the DT. Note that both measures of the training time were performed on a laptop computer as it was not possible to run the DT on the Android device. The results show that NB only takes a few seconds to build a model while the DT algorithm takes on average about 10 minutes for the locomotion task and 2 minutes for the gestures task. The greater training time for the locomotion task results from the larger number of training records in this task.

6.4 Accuracy over time

To further analyse the accuracy results of NB and since we are proposing an incremental approach, we decided to measure how the accuracy changes over time as more records are processed.

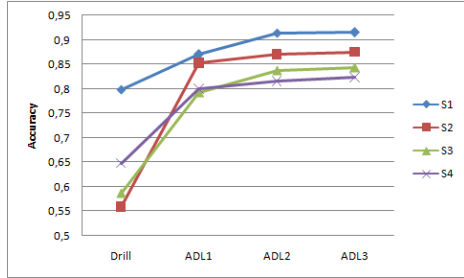


Fig. 2: Accuracy for the Locomotion task

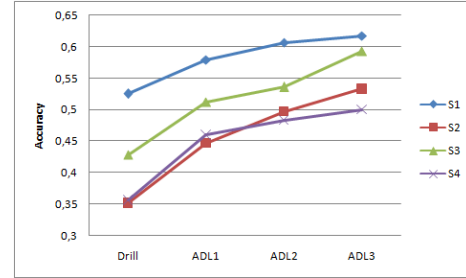


Fig. 3: Accuracy for the Gestures task

Figure 2 shows for the different subjects the accuracy for the locomotion task anytime model after training is performed with data from the different sessions (on the horizontal axis) in an incremental way. The accuracy curves show that the accuracy increases as more data is processed. The results are similar across the subjects, however, it can be observed that the Drill session for subject one is enough to achieve accuracy in the order or 80% while for the other subjects at this stage accuracies of around 60% are obtained. Moreover, the gain in accuracy from

an additional session decreases with the number of sessions as can be observed in Figure 2. This is an interesting result because it can be used to determine when a stable accuracy value has been obtained and no further training is needed.

Figure 3 shows for the different subjects the accuracy for the gestures task anytime model. Again the accuracy curves show that the accuracy increases as more data is processed. However, for this task, which is more demanding than the locomotion one, for the reasons that have been mentioned previously, the accuracy grows more slowly as more records are observed. We should note again that each session has less training data for the gestures task than the locomotion one, which can itself have an impact on the accuracy growth rate with additional sessions. For this task the learning curves are even more similar among the subjects.

6.5 Memory consumed by the model

Since memory is a critical resource in mobile environments, we measured the memory size consumed by the classification models.

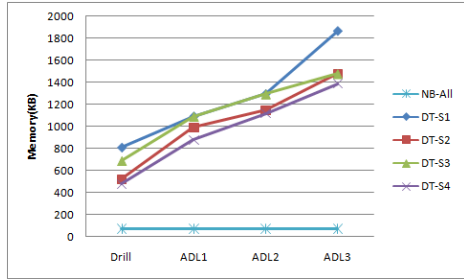


Fig. 4: Memory user in the Locomotion task

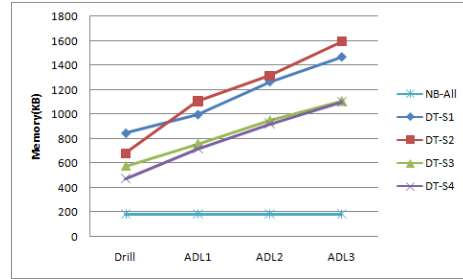


Fig. 5: Memory used in the Gestures task

Figure 4 shows the memory consumption of the incremental NB and DT with the number of training sessions for the locomotion task. Since DT is a non-incremental approach the algorithm was run again with the sessions desired. For instance the results for ADL3 of a particular subject represent a classification model that was built using a training set that contains session ADL3 and the all the previous sessions (i.e., Drill, ADL1 and ADL2). The results show that the NB algorithm resulting model achieves the lowest memory consumption (71KB) and that this value is independent of the number of training records. This is the consequence of the model representation of the NB which only requires to store estimators for the marginal and conditional distributions, which once built stays the same over the entire learning process (possible infinite number of training records). In contrast, the decision tree structure size in memory depends on the tree itself. In Figure 4 we observe that this value often increases linearly with the number of training records.

Figure 5 shows results similar to the Figure 4. For the gestures task the NB model requires again the smallest memory size (185KB). This size is larger than the locomotion model, as this task has more classes, and therefore more estimators need to be stored. For the DT we observe that the trees are of similar size than the ones used to model the locomotion task as the number of classes does not influence the model size directly.

In general these results show that both approaches achieve model sizes that are very small when compared to nowadays mobile devices memory capabilities. However, since the DT requires that the training set must be stored in memory in order to built the model, its usability for on-board modelling is compromised. Consequently, for this reason the majority of studies that use DT for mobile AR require that the model is built on an external server that is later deployed on the mobile for classification of the activities.

6.6 Battery consumed by the process

In this experiment the training process was programmed to be in an infinite loop and was left running in background. In addition to the battery level we measured the processing rate, that is, how many records per second the algorithm processes. This way we can analyse how the processing rate influences the battery consumption and assess the feasibility of the approach. Please note that there a battery cost associated with the data collection from sensors while here we are accessing the file system. Nevertheless, we are controlling our experiment for the impact that the learning algorithm has on the battery, since the cost associated with collecting data from the sensors will be independent of the learning method and will be an existing factor to consider in every approach.

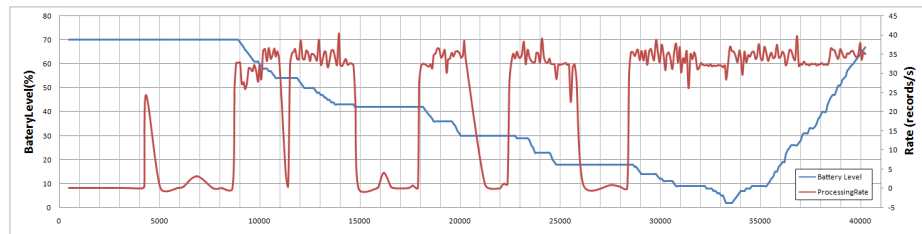


Fig. 6: Battery consumption and processing rate over time

Figure 6 shows the battery consumption and processing rate over a period of about 12hours. The battery level starts at 70% and it decreases slowly over time until the phone is plugged-in at around time 33569s. It can also be seen that the battery consumption rate is related to the processing rate.

The processing rate can be as low as 1 record for every 10 seconds to 40 records per second. This is well within what is required in mobile AR. For instance in [6] where 10-second intervals are considered for basic locomotion activities or

in the more demanding case of the OPPORTUNITY AR challenge dataset [8] used in our experiments, where for the gestures task the shortest gesture found in the dataset lasts about 0.5 seconds. It is considered that such rate is enough to achieve high speed of recognition without missing activities. We should note that the dataset used in the experiments contains a large number of attributes (i.e., 114 numeric attributes + 2 discrete class attributes) than what is normal and no feature selection or tuning to increase the efficiency of our proposed approach. This way we intend to demonstrate the feasibility of the approach in a highly demanding learning task.

A possible way to save battery would be to adapt the processing rate to the situation (i.e., resources and context) [3] to address the mobile application AR requirements.

6.7 Adaptation to different subjects

In MARS, there is greater opportunity for personalisation of the model as the training data is subject specific. Nevertheless, in some applications, for instance in elderly monitoring, asking the subjects to annotate their activities can become an issue if they suffer from Alzheimer or other memory related condition.

In this set of experiments we tested how a model built using Naive Bayes with data from other subjects can be used to accurately classify the activities of a particular subject.

The results indicate (tables not included due to space limitations) that in general for subjects 1, 2 and 3 the models are able to still achieve good accuracy (in both tasks) in relation to the scenario where subject-specific training data is used. Moreover, we observe that for subjects 2 and 3, when the training data from subject 1 is used better accuracy than with the subject-specific data is obtained. We can note that this can be attributed to the fact that the training set for subject 1 has more training records than the other two. Nevertheless, for subject 4 the adaptation is poor, either when its data is used for training or testing. The bad performance with subject 4 can be attributed to the noise that its test dataset includes or maybe to a very different user profile. To further investigate this issue we performed the same experiments without using training data of subject 4 and this results in higher accuracy values.

7 Conclusions and Future Work

In this paper we proposed and experimentally evaluated the Mobile Activity Recognition System (MARS), that is an ubiquitous data stream mining approach to mobile activity recognition. In MARS, a data stream classification algorithm (incremental Naive Bayes) is used to update an anytime model from a stream of ubiquitous sensor data. The main contribution of this work is to show the feasibility to execute such integrated learning approach to mobile AR on the mobile device itself. The advantages of on-board mobile data stream mining are higher personalisation of the AR models (that are built based on individual

users annotated data), increased privacy as the data is not sent to an external site, and also using an adaptive anytime model instead of a static model enables adaptation of the activity profile to changes.

In future work, in line with the latest experiments conducted in this work we plan to study how to depend less on labelled/annotated data, for instance by using a semi-supervised or active learning approach to AR.

Acknowledgements

We would like to thank Dr. Daniel Roggen for the release of the OPPORTUNITY AR challenge dataset labels promptly after the challenge results were presented at the SMC2011 conference workshop. This research is partially financed by project TIN2008-05924 of Spanish Ministry of Science and Innovation. We would also like to thank to the FCT project Knowledge Discovery from Ubiquitous Data Streams (PTDC/EIA-EIA/98355/2008).

References

1. L. Bao and S.S. Intille. Activity recognition from user-annotated acceleration data. *Pervasive Computing*, pages 1–17, 2004.
2. J. Bartolo Gomes, S. Krishnaswamy, M. Gaber, P.A.C. Sousa, and E. Menasalvas. Mars: a personalised mobile activity recognition system. In *Proceedings of the International Conference on Mobile Data Management, MDM 2012, Bengaluru, India, 23-26 July*. IEEE, 2012.
3. J. Bartolo Gomes, E. Menasalvas, and P.A.C. Sousa. Situation-aware data stream mining service for ubiquitous applications. In *Mobile Data Management (MDM), 2010 Eleventh International Conference on*, pages 360–365. IEEE, 2010.
4. J. Gama, R. Sebastiao, and P.P. Rodrigues. Issues in evaluation of stream learning algorithms. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 329–338. ACM New York, NY, USA, 2009.
5. S. Krishnaswamy, J. Gama, and M.M. Gaber. Advances in data stream mining for mobile and ubiquitous environments. In *Proceedings of the 20th ACM international conference on Information and knowledge management*, pages 2607–2608. ACM, 2011.
6. J.R. Kwapisz, G.M. Weiss, and S.A. Moore. Activity recognition using cell phone accelerometers. *ACM SIGKDD Explorations Newsletter*, 12(2):74–82, 2011.
7. S Preece, Jy Goulermas, Lpj Kenney, D Howard, K Meijer, and R Crompton. Activity identification using body-mounted sensors a review of classification techniques. *Physiological Measurement*, 30(4):R1–R33, 2009.
8. Hesam Sagha, Sundara Tejaswi Digumarti, Jos del R. Milln, Ricardo Chavarriaga Lozano, Alberto Calatroni, Daniel Roggen, and Gerhard Trster. Benchmarking classification techniques using the Opportunity human activity dataset. In *IEEE International Conference on Systems, Man, and Cybernetics*, 2011.