Smartphone-based Activity Recognition using Hybrid Classifier
Utilizing Cloud Infrastructure for Data Analysis

Bingchuan Yuan, John Herbert and Yalda Emamian
Department of Computer Science, University College Cork, Cork, Ireland

Keywords: ADLs, Smartphone, Wearable Wireless Sensor, Machine Learning, Cloud Infrastructure, Unsupervised Learning, Real-time Activity Recognition.

Abstract: Learning and recognizing the activities of daily living (ADLs) of an individual is vital when providing an individual with context-aware at-home healthcare. In this work, unobtrusive detection of inhabitants' activities in the home environment is implemented through the smartphone and wearable wireless sensor belt solution. A hybrid classifier is developed by combining threshold-based methods and machine learning mechanisms. Features extracted from the raw inertial sensor data are collected from a Body Area Network (BAN) (consisting of the Zephyr BioHarness sensor and an Android smartphone), and are used to build classification models using different machine learning algorithms. A cloud-based data analytics framework is developed to process different classification models in parallel and to select the most suitable model for each user. The evolving machine learning mechanism makes the model become customizable and self-adaptive by utilizing a cloud infrastructure which also overcomes the limitation of the computing power and storage of a smartphone. Furthermore, we investigate methods for adapting a universal model, which is trained using the data set of all users, to an individual user through an unsupervised learning scheme. The evaluation results of the experiments conducted on eight participants indicate that the proposed approach can robustly identify activities in real-time across multiple individuals: the highest recognition rate achieved 98% after a few runs.

1 INTRODUCTION

As health awareness increases and technology advances, there is an increasing demand for an automated healthcare infrastructure that provides round-the-clock assessment of a subject's well being (Singer, 2011). Remote healthcare monitoring in a Smart Home is a potential solution allowing an elderly person to remain living safely and independently. A Smart Home environment usually consists of three main components: pervasive sensors, an intelligent reasoning system and actuators. Sensors are low level devices that capture changes in the home environment. Data streamed from sensors can be interpreted by the reasoning system for rapid detection of events in real-time. Certain event information can then be sent over the Internet to a Remote Monitoring Server and intelligent decisions can be made to allow the actuators to adjust the environment. This kind of infrastructure is the next generation of personal home healthcare.

Our previous work (Yuan and Herbert, 2011b; Yuan and Herbert, 2011a; Yuan and Herbert, 2012) describes *CARA* (Context-aware Real-time Assistant) an intelligent system especially designed for pervasive healthcare. The ability to accurately recognize and continuously monitor activities of daily living (ADLs) is one of the key features that *CARA* is expected to provide. ADL is associated with both physical and mental health and is a primary indicator of quality of life (Skelton and McLaughlin, 1996). Indeed, some age-related diseases(cognitive impairments, mild dementia and Parkinson's disease) have a direct impact on the ADL of the elderly (White et al., 2001). Activity recognition has been studied as part of a healthcare solution to reduce the necessity for carer supervision of individuals. Activity recognition can also be used in conjunction with pattern recognition to detect changes in a subject's routine.

In this paper, a novel and robust ADL recognition approach is developed and evaluated. The solution involves identifying a user's activity through the combined use of inertial sensors(accelerometer and gyroscopes) built-in to the smartphone along with a wearable wireless sensor attached to the chest. Raw sensor data are collected, filtered and extracted into different features. These features are then used to build and update classification models using several machine

Yuan B., Herbert J. and Emamian Y..
Smartphone-based Activity Recognition using Hybrid Classifier - Utilizing Cloud Infrastructure for Data Analysis.
DOI: 10.5220/0004723900140023
In *Proceedings of the 4th International Conference on Pervasive and Embedded Computing and Communication Systems* (PECCS-2014), pages 14-23
ISBN: 978-989-758-000-0

learning algorithms. The most remarkable part of our approach is that it is able to refine the classification model through a cloud-based data analytics framework so that it becomes customizable and adaptive to different individuals. As a hybrid system, a threshold based method is used to distinguish static and dynamic activities and a rule-based reasoning mechanism is applied to identify simple static activities and a machine learning classifier is used to classify complex dynamic activities which significantly improves the cost efficiency under the limitation of computing power of a mobile device. Furthermore, by combining the wearable wireless sensor and the smartphone, the average recognition rate of ADL can achieve over 95% in a real usage environment.

2 STATE OF THE ART AND RELATED WORK

Advances in ubiquitous and pervasive computing have resulted in the development of a number of sensing technologies for capturing information related to human physical activities. Two approaches for activity recognition have been extensively studied using different underlying sensing mechanisms. The first method relies upon environmental sensors (e.g. RFID tags, cameras) to track aspects such as motion, location and object interaction (Tapia et al., 2004). It is a promising approach for recognizing activities, but the major hurdle in implementing these systems outside of trials is how intrusive these sensors are. Usually a large investment is involved in setting up and maintaining the system, and in some cases it is only feasible for use in laboratory conditions.

The second method uses a Body Area Network (BAN) to track the acceleration of specific limbs as well as the body as a whole. Some of the existing work on wearable sensor based activity recognition uses several accelerometers placed on different parts of the body (Tapia et al., 2007). Other research has explored the use of multiple kinds of wearable sensors for activity recognition (e.g. accelerometers, temperature sensors and microphones)(Maurer et al., 2006). Wearable sensors are often non-invasive, unobtrusive and require less effort to set up. Nevertheless, most of the wearable sensors are required to be fixed onto special locations on the subject's body which makes it inconvenient and impractical for continuous long term monitoring. While the ubiquity of smartphones and their capability to support sensor data collection, processing and communications make them an alternative platform to wearable sensors. Recently, researchers have been working on using smartphones as mobile sensing and computing platforms to achieve activity recognition (J.R. Kwapisz and Moore, 2010; Khan et al., 2010; Stefan et al., 2012). However, the performance of these approaches is less effective due to the ambiguity of upper body movement tracking. To improve it, our approach makes use of two sets of sensors; instead of using a single smartphone an additional Zephyr BioHarness sensor was used to monitor trunk movement. The recognition rate was dramatically improved by this combination, as shown in the evaluation results.

On the other hand, activities of daily living in the home environment can be generalized into two categories. Simple motionless activities, such as sitting, standing, lying and bending, correspond to a static posture of the human body. These activities may be mostly recognized by using a threshold based method (Lyons et al., 2005). Complex dynamic activities, such as walking stairs, cooking, sweeping and washing hands, involve multiple overlapping actions. These activities can be recognized by identifying patterns of people's movement (Ravi et al., 2005). We build on these approaches and extend them to test the ability of real-time identification of daily activities through the combination of the aforementioned methods.

Using a smartphone as a primary device for data collection and processing increases the likelihood of data coverage and represents a minimal maintenance commitment and cost to the user. However, there is a limitation of the hardware resource in a smartphone when processing heavy machine learning tasks. Fortunately, the emerging of cloud computing could solve the problem. As a delivery model for IT services, cloud computing offers a cost-effective way to support data analysis technologies with high flexibility, scalability and availability for accessing data, discovering patterns, and deriving models. There are lots of ongoing research on mobile cloud computing, and a wide range of potential mobile cloud applications have been recognized in the literature (Huerta-Canepa and Lee, 2010; Frederking and Brown, 2010; Marinelli, 2009). These applications fall into different areas such as image processing, natural language processing, multimedia search and sensor data management. In this work, we focus on introducing a cloud-based data analytics framework especially designed to refine classification models for activity recognition using multiple machine learning methods. By utilizing the cloud infrastructure, the smartphone, even with limited computational resources, can perform intelligent real-time classification and this provides novel functionality in our solution.

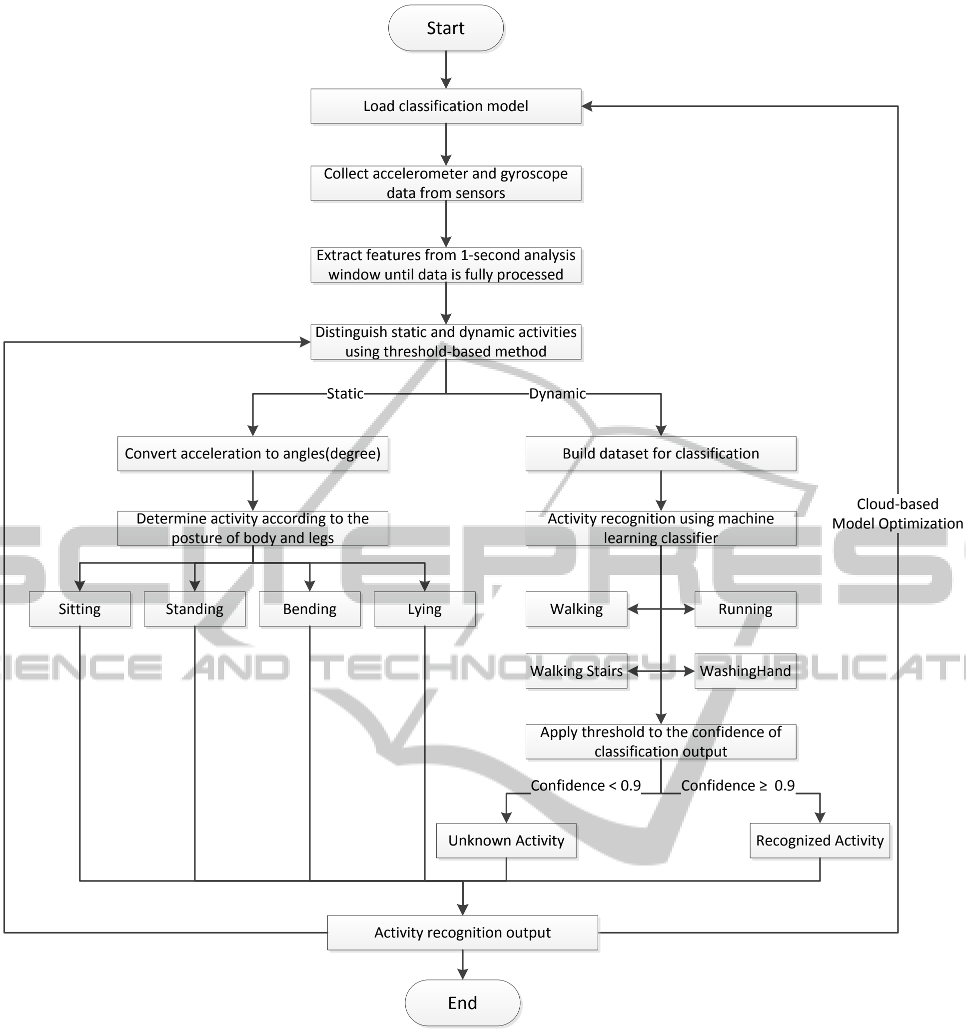


Figure 1: Flowchart of activity recognition.

3 MATERIALS AND METHODS

3.1 Activity of Daily Living

Previously, we have developed a reasoning framework for CARA pervasive healthcare that can help extend independent living for the elderly in a smart home environment by monitoring the person and ambient changes to detect anomaly situations (Yuan and Herbert, 2012). For a smart home based monitoring system, it is important to detect human body movement which can provide the basic activity context. By combining activity context along with other environmental contexts, the context-aware monitoring system is able to perform better reasoning for healthcare. In this paper, we focus on introducing how the system recognizes human activities through BAN without assistance of any smart home sensors. The basic activity of daily living can be divided into two categories: static posture and dynamic movement. Static posture indicates the position of the body which consists of ***Sitting, Standing, Lying, Bending and Leaning back***, whereas dynamic movement is the compilation of a series of multiple actions. The dynamic activities we considered in our system include: ***Walking, Running, Walking Stairs, Washing Hands, Sweeping, Falling***. A custom-design Android application for activity recognition has been implemented and run on Samsung Galaxy III. The control flow is as follows:

1. The machine learning classification model is loaded from the cloud.
2. Raw data are collected from BioHarness sensors

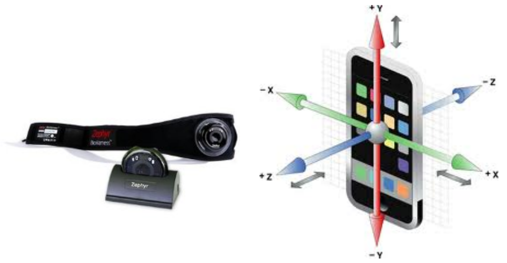
and smartphone build-in accelerometer and gyroscope sensors.

3. The signals are then low-pass filtered and a 1s window is moved over the signal and overlapped every 500ms.
4. The features corresponding to each window are extracted.
5. Static and dynamic activities are distinguished using a threshold-based method where a threshold value of the acceleration is applied.
6. Static activities are divided into sitting, standing, bending, lying and leaning back by applying threshold angles for both trunk and thigh.
7. Dynamic activities are classified using machine learning classifier based on the dataset of extracted features.
8. Activities are labelled as correctly classified if the output confidence is high enough.
9. Features and activity label of each detected case are stored in the data file.
10. Recognized activity is used to retrain the classifier and update the classification model using the cloud-based data analysis framework.

The process of activity recognition is summarised in Fig. 1 and further details are described in the following sections.

3.2 Data Collection

The data were collected by using a Samsung Galaxy SIII mobile phone (Samsung Inc., 2012) and wearable Zephyr BioHarness sensor as shown in Fig. 2(a) (Zephyr Inc., 2011). The embedded tri-axial accelerometer and gyroscope sensor measure the 3D-acceleration and orientation of the smartphone. The three axes of acceleration are dependent upon the orientation of the phone, the x-axis runs parallel to the width of the phone, the y-axis runs the length of the phone, and the z-axis runs perpendicular to the face of the phone, as shown in Fig. 2(b). The sensor integrated into the mobile device is easy to use without assistance and can be carried comfortably for long periods of time (Choudhury and Consolvo, 2008). The data collection was done by performing experiments on eight postgraduate students at University College Cork, Ireland. Subjects were asked to perform a series of activities while carrying the smartphone in the front pocket of their trousers and the BioHarness sensor on the chest. The BioHarness sensor data were transmitted to the smartphone through a Bluetooth connection in real-time. We created an application running on a



(a) Zephyr BioHarness (b) 3D-acceleration

Figure 2: Body area network for sensor data collection.

tablet device that communicates with the smartphone while the subject performed the activities. The tablet worked as a monitor and a controller that displays the label of the ongoing activity and controls the phone to start or stop recording activity data. 9 channels of sensor readings, 3D-acceleration and orientation of the thigh and 3D-acceleration of the trunk with associated timestamps *(ThighACCx, ThighACCy, ThighACCz, ThighGYROx, ThighGYROy, ThighGYROz, TrunkACCx, TrunkACCy, TrunkACCz)* were collected and subsequently used for the evaluation of the activity classification algorithms. Fig. 3 illustrates the sample of raw sensor data collected by the smartphone.

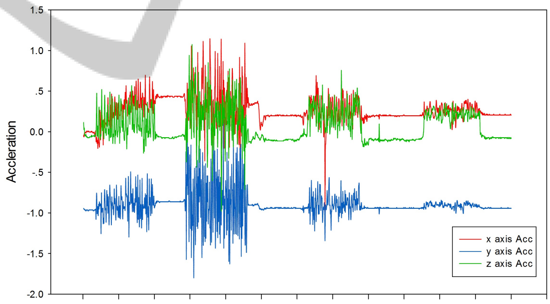


Figure 3: Plot of raw 3-axis acceleration of the smartphone.

3.3 Feature Extraction

As to the sampling frequency, the sensors in the Android phone trigger an event whenever the accelerometer or gyroscope values change. The rate of events can be set to one of four thresholds: *fastest, game, normal*, and *UI*, with *fastest* being the fastest sampling rate and *UI* being the slowest. In order to balance the speed of data processing and activity feature extraction, the sampling rate for the experiment was set to *normal* which is about 15Hz and we found it is fast enough for the daily activity classification.

Standard classifiers do not work well on the raw sensor data. It is essential to transform the raw data into a representation that captures the salient features of the raw data. This is typically performed by break-

Table 1: Description of features extracted from raw sensor data.

Feature	Trunk Accelerometer	Thigh Accelerometer	Thigh Orientation
Min	X, Y, Z, Peak ACC	X, Y, Z, Absolute ACC	Azimuth, Pitch, Roll, Absolute GYRO
Max	X, Y, Z, Peak ACC	X, Y, Z, Absolute ACC	Azimuth, Pitch, Roll, Absolute GYRO
Mean	X, Y, Z, Peak ACC	X, Y, Z, Absolute ACC	Azimuth, Pitch, Roll, Absolute GYR
Standard Deviation	X, Y, Z, Peak ACC	X, Y, Z, Absolute ACC	Azimuth, Pitch, Roll, Absolute GYRO
Zero Cross	X, Y, Z	X, Y, Z	Azimuth, Pitch, Roll
Mean Cross	Peak ACC	Absolute ACC	Absolute GYRO
Angular Degree	X, Y, Z	X, Y, Z	

ing the continuous data into windows of certain duration. In this work, we experimented with one second time window which is overlapped by one half of the window length. Hence, each window is a single instance, but any given data point contributes to two instances. This method has been shown to be effective in earlier work using accelerometer data (Bao and Intille, 2004). Table 1 summarized a number of features extracted from each window. Each window was represented as a feature vector of length 66. The processed data were saved in the smartphone in an arff(Attribute-Relation File Format) file for data analysis.

3.4 Threshold-based Mechanism

Among all the features, the standard deviation of acceleration indicates the variability of the accelerometer signal for each 1-s window of recorded data. High variability would be expected during dynamic activities whereas static activities result in low variability. Thresholds are then applied to the standard deviation of both the smartphone and BioHarness accelerometer signals. If either signal is above the threshold for that second the activity is considered dynamic and if both of them are below the threshold, the activity is deemed static. Thresholds were determined empirically from the collected data. The threshold for trunk acceleration was set at 0.25 m/s^2 and threshold for thigh acceleration was set to 0.2 m/s^2 to ensure that all motionless activities are detected. Fig. 4 shows an example of the standard deviation threshold for BioHarness accelerometer readings.

$$\theta_{degrees} = \frac{180}{\pi} \arccos\left(\frac{a}{g}\right) \qquad (1)$$

When the activity is deemed static, the mean accelerations over the one second window are converted to a corresponding inclination angle using the arc cosine transformation of Eq. 1, where a is the mean acceleration, g is the gravity of earth, θ *in degrees* corresponds to the angles of the trunk and thigh. Specific trunk and thigh inclination ranges are set based

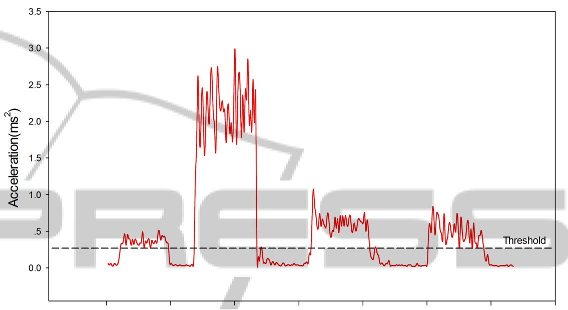


Figure 4: Standard deviation threshold for trunk accelerometer signals.

on the finding of Lyons et al. (Lyons et al., 2005). They proposed the *best estimate* threshold which accurately reflect real-life trunk and thigh ranges. However, they only considered three static activities, sitting, standing and lying. In our work, we added two more static activities, bending and leaning back, for classification. The ranges of thigh and trunk angles

Table 2: Threshold in degrees for thigh and trunk posture detection.

Thigh Posture	Upper Thres.	Lower Thres.
Vertical	0	45
Horizontal	45	90
Trunk Posture	**Upper Thres.**	**Lower Thres.**
Vertical	-30	30
Bending	30	60
Horizontal	$\lvert\pm 90\rvert$	$\lvert\pm 60\rvert$
Leaning Back	-60	-30

for each posture are listed in Table 2. Static activity is classified according to the posture of trunk and thigh using rule-based methods. In the trial, detection accuracies of 98% and greater were achieved for each of the static activities.

3.5 Machine Learning Classifier

To recognize dynamic activities such as walking, walking stairs and running, the machine learning classifier was trained to produce the classification model. The Weka machine learning package (Witten et al., 2011) was used in this study for the purposes of developing the machine learning mechanism for the activity detection. This package provides a collection of machine learning algorithms for data mining tasks which can be used in the android application. Four different machine learning algorithms were investigated and evaluated in this work:

- **Bayesian Network.** A probabilistic graphical model using estimator classes and quality measures.
- **Decision Tree.** Generates a decision tree and navigates from the root of the tree down to a leaf for classification.
- **K-Nearest Neighbours.** An instance-based classifier, classification is based on some similarity function.
- **Neural Network.** A multiple layer of perceptron that uses back propagation to classify instances

We used the default parameters associated with each of the classifiers. The classifier obtains a model during the training stage. After training, it can predict the class membership for new instances using the classification model. Based on our experiments we have found that the performance of different classifier varies from person to person which mostly depends on the training dataset collected from each individual. But overall, the Decision Tree algorithm has provided more comprehensive results in terms of accuracy and efficiency in comparison with the other algorithms. The details of the experiment results are discussed in Section 4.

3.6 Cloud-based Model Optimization

Data collected from all the subjects were pooled together to build a universal classification model, referred to as the *Default Model* in the rest of the paper. It works as the default for each user only if there is no existing personal model available for that user. However, it is not the suitable model for a specific individual since the model is trained and evaluated using the data gathered from all users; the body and behaviour pattern of each user is different. In order to improve the classification performance efficiently, we introduced the idea of model adaptation for the optimization of the machine learning classifier by utilizing the cloud infrastructure.

As shown in Fig. 5, the principle of model optimization is to keep updating the classification model for an individual user while the activity recognition task is carrying on. All the new users start with a default classification model, which in turn gets refined and adapted to each individual user for better performance when more activity data are available as users carry the phone.

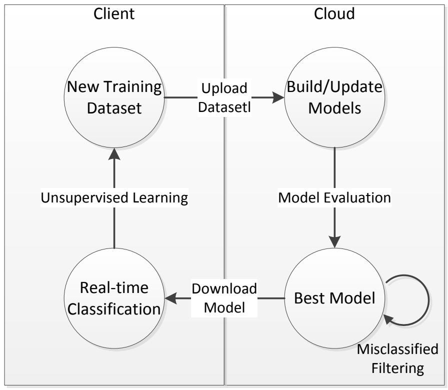


Figure 5: The process of model optimization.

On the client side, real-time activity recognition is carried out in the smartphone using the classification model. An unsupervised learning scheme was applied to generate new training data which is based on self-training using unlabelled data without any user input. It reuses the predicted label and confidence statistics generated by the machine learning classifier during the inference process to select new training samples. The adaptation method determines whether a data sample is suitable for adaptation according to the confidence level of the inference result and uses high confidence samples. The new training data are recorded in a data file in the smartphone and are uploaded to the cloud storage periodically.

On the cloud side, multiple worker roles are deployed for analysing user's activity data.

- **Controller Node.** Control and manage the flow of incoming data and add the new messages in to the Task Queue for further processing.
- **Machine Learning Nodes.** Each of the worker roles reads a message from its own Task Queue and starts producing a model based on a different machine learning classifiers.
- **Universal Node.** A special machine learning worker role that is designed to deal with data from all the users to update the Default Model using a dedicated classifier (e.g. Neural Network).
- **Evaluation Node.** Select the most suitable model for a specific user by comparing the evaluation results of each classification model. Moreover, a misclassified filter is also applied to the user's data after the best model is found.

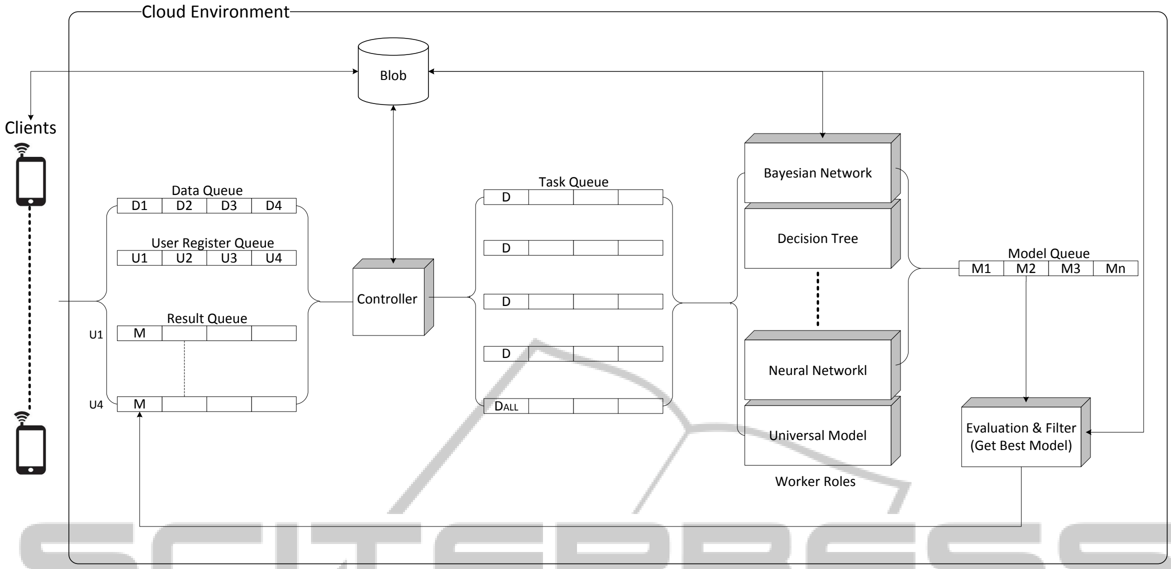
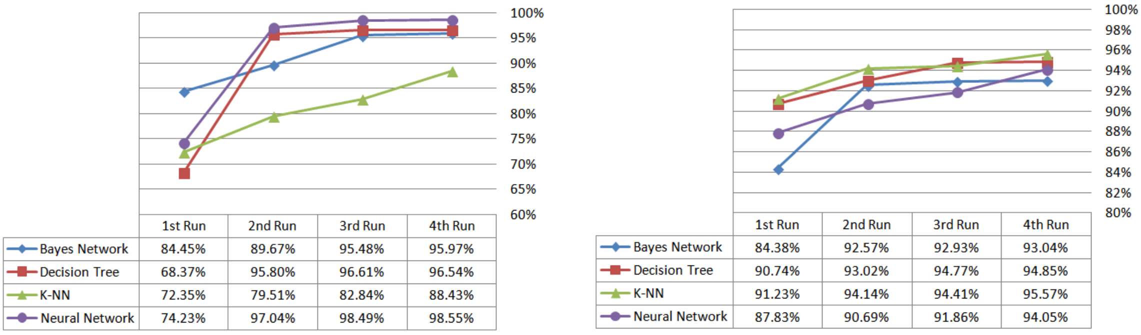


Figure 6: Structure of cloud-based data analysis framework.

The process of classification model optimization is illustrated in Fig. 6. The URL of the training data are collected by the Controller node and assigned as different tasks to machine learning worker roles. The data are then used to either build new classification models in the case of a new user or refine the existing models. Moreover, the data is also fused with the available data from all users to update the *Default Model* for new user. In the experiment, four worker roles worked in parallel to process machine learning tasks. Each of the worker roles is designed to handle one machine learning algorithm. The number of worker roles is able to be adjusted on demand which makes use of the flexibility and scalability of the cloud infrastructure. The models are generated and evaluated for each machine learning classifier in worker roles. After gathering the evaluation results of all the models in the Evaluation node, the most suitable model is selected according to the performance of each model. The data (combining new training data and any previously existing data) of that user are then filtered by a misclassified filter using the best model and stored in the cloud storage as previous data. Once the process is finished, the best model is sent back to the user automatically. Thus, the new or updated model can be used in the client for real-time classification.

4 EXPERIMENTS AND RESULTS

Eight volunteers (five male, three female) were involved in our experiments to help us evaluate the feasibility of the proposed approach. The activity data were collected in a home setting and recorded in data files for training and testing machine learning classifiers. For supervised learning, subjects were asked to perform static and dynamic activities sequentially while carrying the smatphone and wearing the body sensor. A controller application running on an Android Tablet remotely controlled the smartphone to start or stop recording activity data and to label the recorded data on the fly. Firstly, we built the *Personalized Model* which is a completely user dependent approach. It requires training machine learning classifier on each individual user's activity data and generates a user dependent model for each user. Clearly, this scheme is superior to other models in terms of performance for a specific user, but its lack of usability and scalability greatly limits its application. In addition, the supervised learning scheme requires a great effort of manually labelling the activity data which is a tedious and heavy task for the user. To improve it, we decided to move on to a semi-supervised approach where a universal classifier is trained with manually labelled data from all users (called the *Default Model*). As we discussed in the last section, an unsupervised learning scheme was applied to gather new training data to update the model so that it gets adapted to each individual user in a progressive manner.



(a) Overall model accuracy for the female user A

(b) Overall model accuracy for the male user B

Figure 7: Average accuracy of each machine learning classifier in four runs.

Table 3: Detail of performance of different classification models.

Classifier	**TP Rate**	**FP Rate**	**Precision**	**Recall**	**F-Score**	**Time (ms)**	**Accuracy**
First Run (1980 instances $\longrightarrow$ 1874 instances)							
Decision Tree	0.684	0.034	0.839	0.684	0.657	1838	68.37%
Bayesian Network	0.845	0.011	0.931	0.845	0.860	2725	84.45%
K-Nearest Neighbour	0.724	0.044	0.811	0.724	0.684	4849	72.35%
Neural Network	0.742	0.040	0.811	0.742	0.720	84682	74.23%
Second Run (3493 instances $\longrightarrow$ 3392 instances)							
Decision Tree	0.958	0.006	0.960	0.958	0.957	1248	95.80%
Bayesian Network	0.897	0.010	0.943	0.897	0.899	1482	89.67%
K-Nearest Neighbour	0.795	0.036	0.860	0.795	0.763	5504	79.51%
Neural Network	0.970	0.004	0.972	0.970	0.970	145111	97.04%
Third Run (5482 instances $\longrightarrow$ 5403 instances)							
Decision Tree	0.966	0.006	0.967	0.966	0.966	1358	96.61%
Bayesian Network	0.955	0.005	0.967	0.955	0.958	1727	95.48%
K-Nearest Neighbour	0.828	0.031	0.873	0.828	0.810	7019	82.84%
Neural Network	0.985	0.002	0.985	0.985	0.985	227774	98.49%
Fourth Run (6790 instances $\longrightarrow$ 6714 instances)							
Decision Tree	0.955	0.007	0.958	0.965	0.965	1692	96.54%
Bayesian Network	0.960	0.005	0.966	0.960	0.962	1357	95.97%
K-Nearest Neighbour	0.884	0.020	0.899	0.884	0.881	9041	88.43%
Neural Network	0.985	0.002	0.986	0.985	0.985	281971	98.55%

The proposed cloud-based approach was tested on two users(one female user A and one male user B). The manually labelled data gathered from these two users were used as the testing set to evaluate four classification models. Each user started with the *Default Model* and was able to obtain his/her *Adapted Model* after the first run. The model was refined and updated while the experiment was carried on. After four runs of each user, the overall accuracy of the classification model was boosted and the best performance achieved was over 95% (see Fig. 7). The detailed results for each classification model obtained from each run of User A are summarized in Table 3. It can be observed that the performance of each model in the first run is quite poor, because the *Default Model* dose not suit the user very well. However, after updating the *Adapted Model* and filtering the misclassified instances in the next few runs, the accuracy of most models remains consistently above 90% except for K-NN. The best accuracy was obtained with the Neural Network method while the most cost efficient model was the Decision Tree. Note that the execution time for building and evaluating the model using the Decision Tree algorithm was around 1.5s on average while the processing time of the Neural Network linearly decreased with the growth of the instances. In this case, the Decision Tree was considered as the best *Adapted Model* for this user. The *Adapted Model* rep-

Table 4: Confusion matrix by using K-NN classifier in the Default Model.

Activity	a	b	c	d	e	f	g	h	i	j	k	l
WALKING (a)	1852	0	10	0	0	0	6	0	0	0	0	0
RUNNING (b)	1	1105	32	1	0	0	0	0	0	0	0	0
WALK STAIRS (c)	47	0	1937	4	0	0	1	4	0	0	2	0
SWEEPING (d)	6	0	1	1378	14	0	4	0	0	0	0	0
WASHING HANDS (e)	0	0	0	0	873	0	7	0	0	2	1	0
FALLING (f)	0	0	2	1	5	32	6	1	2	3	0	0
STANDING (g)	0	0	0	0	0	0	822	0	0	0	0	0
SITTING (h)	0	0	0	0	0	0	0	972	0	0	0	0
LYING (i)	0	0	0	0	0	0	0	1	649	0	0	0
BENDING (j)	0	0	0	0	0	0	0	0	0	718	0	0
LEANING BACK (k)	0	0	0	0	0	0	0	0	0	0	557	0
ROLLING (l)	0	0	4	10	1	1	1	5	8	0	1	187

resents a middle ground between the *Default Model* and the *Personalized Model*; it got refined and eventually yielded 98% accuracy after four runs through model optimization, and it is the model that is actually used in real-time activity recognition.

As for the *Default Model*, it was initially built using the data collected from eight people using a supervised learning scheme and then kept being updated in the cloud once new data uploaded by any user was available. Table 4 presents the confusion matrix for the considered daily activities. The results were obtained using a 10-fold cross-validation method to evaluate the *Default Model* that was built by the K-NN classifier. The matrix shows that most errors occurred between *Falling*, *Standing* and *lying*. Nevertheless, the average accuracy is still over 95% since the original *Default Model* was built through a supervised learning manner. The reason for misclassifying Falling is that it is a quick action which usually involves a series of static pre-actions and post-actions, e.g. Standing and lying. So it is difficult to classify it through a machine learning model using a fixed sliding window method. However, a threshold based method would solve this problem because the change of activity readings are dramatic at the moment people fall. We evaluated this method and the recognition rate of Falling was boosted from 62% to 91%.

Instead of using a fixed length of sliding window, we also investigated our system using dynamic sliding window which automatically determines the start and end of an activity according to the dynamic activity threshold. The performance of using dynamic window is even better, however this method has the limitation that it requires an activity to be completed before it can be recognized. The delay of activity recognition makes it less practical for real-time monitoring, but it shows the potential for improving our recognition system.

5 CONCLUSIONS

In this paper, we proposed a robust system of daily activity monitoring using hierarchical classification by combining rule-based reasoning and multi-class machine learning algorithms. Human daily activities can be naturally represented through hierarchies, such as motion and motionlessness. Firstly, rule-base reasoning was used to separate the sensing data into two groups: static and dynamic. Static activities are identified based on the posture of the body which is calculated from 3D-acceleration, and dynamic activities are classified respectively by using adapted classification models. By utilizing the cloud infrastructure, the system provides high scalability and availability for data analysis and model management. The data processing and classification algorithms are implemented in the smartphone for real-time activity monitoring while the cloud-based data analysis and model evaluation are conducted off-line. The experimental results compare favourably with other work using body sensors (Stefan et al., 2012). Moreover, the performance of our approach shows a significant improvement in comparison to the approach using a single smartphone (Zhang et al., 2010) and the approach built on fixed machine learning algorithms (Andreu and Angelov, 2013). This shows a lot of promise for using smartphones as an alternative to dedicated sensors and using the cloud-based data analytics framework to process machine learning tasks.

A limitation of this study is that the rule-base reasoning depends on two thresholds to separate the static and dynamic activities. Although, the chosen threshold values work well in this study, they may not be generally suitable for other cases and for different sensors. Future work will evaluate new algorithms, which can separate motion and motionless activities automatically, without the need for static thresholds.

REFERENCES

Andreu, J. and Angelov, P. (2013). An evolving machine learning method for human activity recognition systems. *Journal of Ambient Intelligence and Humanized Computing*.

Bao, L. and Intille, S. (2004). Activity recognition from user-annotated acceleration data. *Pervasive Computing*.

Choudhury, T. and Consolvo, S. (2008). The mobile sensing platform: An embedded activity recognition system. *IEEE2008 Pervasive Computing*.

Frederking, R. and Brown, R. (2010). The pangloss-lite machine translation system. In *Proceedings of the Second Conference of the Association for Machine Translation in the Americas*.

Huerta-Canepa, G. and Lee, D. (2010). A virtual cloud computing provider for mobile devices. In *Proceeding of the 1st ACM Workshop on Mobile Cloud Computing & Services: Social Networks and Beyond*.

J.R. Kwapisz, G. W. and Moore, S. (2010). Activity recognition using cell phone accelerometers. In *Proceedings of the Fourth International Workshop on Knowledge Discovery from Sensor Data*.

Khan, A., Lee, Y., Lee, S., and Kim, T. (2010). Human activity recognition via an accelerometer-enabled-smartphone using kernel discriminant analysis. In *5th International Conference on Future Information Technology*.

Lyons, G., Culhane, K., Hilton, D., Grace, P., and Lyons, D. (2005). A description of an accelerometer-based mobility monitoring technique. *Medical Engineering & Physics*.

Marinelli, E. (2009). Hyrax: cloud computing on mobile devices using mapreduce. Master's thesis, Carnegie Mellon University.

Maurer, U., Samilagic, A., Siewiorek, D., and Deisher, M. (2006). Activity recognition and monitoring using multiple sensors on different body positions. In *IEEE Proceedings on the International Workshop on Wearable and Implantable Sensor Networks*.

Ravi, N., Dandekar, N., Mysore, P., and Littman, M. L. (2005). Activity recognition from accelerometer data. In *Proceeding of the National Conference on Artificial Intelligence*.

Samsung Inc. (2012). Android smartphone: Samsung galaxy siii. Description Available online at http://en.wikipedia.org/wiki/Samsung_Galaxy_S_III.

Singer, E. (2011). The measured life. In *Technology Review*, volume 114.

Skelton, D. and McLaughlin, A. (1996). Train functional ability in old age. Physiotherapy.

Stefan, D., Das, B., Krishnan, N., Thomas, B., and Cook, D. (2012). Simple and complex activity recognition through smart phones. In *8th International Conference on Intelligent Environments*.

Tapia, E., Intille, S., Haskell, W., Larson, K., and Friedman, R. (2007). Real-time recognition of physical activities and their intensities using wireless accelerometers and a heart rate monitor. In *Proceedings of the 11th IEEE International Symposium on Wearable Computers & Physics*.

Tapia, E., Intille, S., and Larson, K. (2004). Activity recognition in the home setting using simple and ubiquitous sensors. In *Proceedings of PERVASIVE*.

White, D., Wagenaar, R., and Ellis, T. (2001). Monitoring activity in individuals with parkinson's disease: A validity study. *Jurnal of Neurological Physical Therapy*, 30.

Witten, H., Frank, E., and Hall, A. (2011). *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, San Francisco, 3rd edition edition.

Yuan, B. and Herbert, J. (2011a). Non-intrusive movement detection in cara pervasive healthcare application. In *The 2011 International Conference on Wireless Networks*.

Yuan, B. and Herbert, J. (2011b). Web-based real-time remote monitoring for pervasive healthcare. In *IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops)*.

Yuan, B. and Herbert, J. (2012). Fuzzy cara - a fuzzy-based context reasoning system for pervasive healthcare. In *Procedia Computer Science*, volume 10.

Zephyr Inc. (2011). Wireless professional heart rate monitor & physiological monitor with bluetooth. Description Available online at http://www.zephyr-technology.com/products/bioharness-3/.

Zhang, S., McCullagh, P., Nugent, C., and Zheng, H. (2010). Activity monitoring using a smart phone's accelerometer with hierarchical classification. In *6th International Conference on Intelligent Environments*.