EVALUATING STATE-OF-THE-ART CLASSIFIERS FOR HUMAN ACTIVITY RECOGNITION USING SMARTPHONES

Athanasios Lentzas*, Andreas Agapitos †, Dimitris Vrakas ±

**Department of Informatics Aristotle University of Thessaloniki, Greece, alentzas@csd.auth.gr, † Department of Informatics Aristotle University of Thessaloniki, Greece, andragap@csd.auth.gr, ± Department of Informatics Aristotle University of Thessaloniki, Greece, dvrakas@csd.auth.gr*

Keywords: Human Activity Recognition, smartphones, wearables, deep learning, instance-based learning.

Abstract

Human activity recognition using smartphones and wearables is a field gathering a lot of attention. Although a plethora of systems have been proposed in the literature, comparing their results is not an easy task. As a universal evaluation framework is absent, direct comparison is not feasible. This paper compares state-of-the-art classifiers already used on mobile human activity recognition, under the same conditions. In addition, an Android application was developed and the method yielding the best results was evaluated in real world in a semi-supervised environment. Results shown that deep learning techniques have better performance and could be transferred to a phone without many modifications.

1 Introduction

Human Activity Recognition (HAR) is the task of correctly identifying human actions and activities, given a sensory input. The input can originate from various sources, such as smartphones, wearables, ambient sensors etc. In general, sensors can be divided into two categories: external and wearables. The former refers to fixed location sensors (ambient sensors, Radio Frequency Identification, cameras etc.) and the latter to wearable devices. Applications can vary from elder and youth care (e.g., infants sleep monitoring) to athlete monitoring.

With smartphones and wearables becoming available to more people as well as being equipped with accelerometer and gyroscope sensors, they have become a solid choice for data gathering [1]. Although the use of those sensors can potentially limit the number of recognized activities, their low cost, data processing capabilities and people's familiarity with those devices, make them a popular choice. A problem with systems exploiting those technologies is the lack of a common evaluation framework, as well as experiments on real life scenarios with multiple devices.

Direct comparison between different approaches is a dubious task. The use of different datasets, each with its own sampling and filtering techniques, as well as the different locations the sensors are placed on the body, provide divergent accelerometer and gyroscope data. Additionally, each subject used for data gathering have a distinct movement profile, resulting in subject specific data. A taxonomy proposed in [2] attempted to address that problem. Authors took into account design and implementation choices and disregarded all the already mentioned aspects of a HAR system.

Another issue is that each proposed method can recognize a different set of activities. There are activities that have a similar accelerometer and gyroscope profile, such as going up and down the stairs, standing, sitting etc. This can potentially lead to lower performance on systems recognizing similar actions. On par with that, classifying fewer and more distinct activities (i.e. mobility/immobility as presented in [3]) results in better performance.

Furthermore, most systems proposed in the literature were only evaluated in laboratory and controlled environments and not in real life. During data gathering each participant performed every action under controlled conditions with definite boundaries. Pre-trained models should be evaluated on persons acting under noisy and uncontrolled environments, where the boundaries and transition between each action are not distinct. Activities performed on different terrain, by persons that were not used for data gathering can result in different sensor reading. In addition to that, while performing experiments on mobile devices, the energy efficiency must be taken into account. The majority of the literature does not take into account the power consumption [4]. This crucial factor has to be considered when using mobile devices, since the power available for filtering and classification are limited.

This paper aims to evaluate state-of-the-art classifiers using the same set of identified activities. Implemented classifiers were trained to recognize the most common set of activities when a smartphone is used (i.e. walking, running, sitting, standing, laying down, climbing stairs). Also, doing a performance comparison of each proposed method, the portability of each method to a new set of data was investigated. Lastly, an android application was developed in order to investigate the performance on real scenarios, in a partially controlled environment.

2 Literature Review

Various techniques have been proposed in the literature for human activity recognition. Apart from statistical models and probabilities, machine-learning approaches are a fast-growing trend in activity recognition.

Decision Trees and Random Forests, an ensemble learning method, have been extensively used in the literature for activity

recognition. In [5], the authors tried to address the problem of the device's orientation by calculating a rotation matrix using 1-second data, gathered with the person standing. All input was then corrected using that rotation matrix. That technique made their approach orientation independent. Similarly, in [6] Random Forests were used not only to identify performed actions, but also to predict the position (i.e. waist, hand etc.) and orientation of the sensor. In [7] several machine-learning algorithms used on activity recognition were evaluated, including Random Forests and Decision Trees. Apart from these, Multilayer Perceptron (MLP), instance-based learning and k-Nearest Neighbors (kNN) were implemented. In their work, experiments were also conducted in order to evaluate the performance of the already mentioned techniques.

Activity recognition using smartphones and wearables, is also used on elderly Ambient Assisted Living. In [8], a fall detection for elders was developed using a threshold method. When acceleration and position variation (derived from gyroscope data) exceed a redefined threshold, an alarm was raised. Using Decision Trees, work presented in [3], identified inactivity based on smartphone data. Immobility tracking is important for elders, as it can be an indicator of health problems.

A combination of smartphones and wrist-worn motion sensors was presented in [9]. Authors evaluated the potential increase on performance when using data originating from two different devices on different body location. Evaluation was done on three different classifiers: Naïve Bayes, kNN and Decision Trees. The combination of different devices could lead to identification of more complex activities, such as drinking coffee, talk, smoke, eating etc.

An increased interested is also observed for Deep Learning application on human activity recognition domain. Recurrent Neural Networks (RNN) and Convolutional Neural Networks (CNN) have been extensively used in the recent literature, not only for data originating from smartphones and wearables, but also from other sources (i.e. cameras, ambient sensors etc.). In [10] a deep CNN was employed, using the accelerometer and gyroscope filtered data without performing any other transformation on them. In [11] a deep RNN approach was proposed and various architectures such as Long Short Term Memory (LSTM) were investigated with promising results.

In [12], a Deep Learning technique was used, along with data fusion from various sensors. Data originating from the magnetic and motion sensors were fused in order to recognize activities of daily living. The Deep Learning approach was evaluated against a feed-forward Neural Network and an MLP, outperforming both.

Further leveraging data fusion techniques, in [13] the authors, presented a complete framework for activity recognition. The method proposed, consisted of 4 different stages: the sensor discovery, data acquisition and processing, data fusion and classifier. As the first module could recognize the available sensors, the proposed solution was not bound to specific devices. That framework could provide a solid choice for mobile activity recognition, removing the need for device and sensor bound solutions.

3 Dataset / Features

As already discussed, it is important to evaluate activity recognition techniques on a common dataset in order to produce comparable results. The dataset employed in this paper is the OPPORTUNITY Activity Recognition Dataset [14], [15] a publicly available dataset released by UCI. The dataset contains tri-axial accelerometer and gyroscope data as well as time and frequency domain features. Data are divided into six different classes, each representing one activity: walking, standing, laying down, walking upstairs, walking downstairs and sitting. According to the authors, a smartphone was placed on the waist of 30 volunteers performing the aforementioned activities. A sliding window with 50% overlap and 128 values fixed-width was used for sampling. Each activity was represented by one window. In order to generate the training and the test subsets, the main set was divided by randomly selecting 21 individuals to form the training set and the rest 7 to form the test set. The procedure used to divide the dataset, ensured that the classifier will not use data from the same subject for training and evaluation. Raw data were denoised using a median filter and a 3rd order low-pass Butterworth filter with 20Hz cutoff frequency.

Extracted time and frequency domain features were normalized and bounded within the [-1, 1] range. A Fast Fourier Transform was employed to extract frequency domain features. For the Decision Tree, Random Forest, Multilayer Perceptron and k-Nearest Neighbors classifier the selected features were the following:

- max, min and average acceleration of each axis
- Average of the square roots of the sum of each axis values
- Standard deviations
- Average of the absolute values of sensor deviations, the fast Fourier transform standard deviations of the spectrum data
- Sum of the squared modulus of the coefficients (fast Fourier transform energy)

The Convolutional Neural Network and the Recurrent Neural Network were implemented with the raw data as input. Six vectors were used with a size of 128, each containing the raw signals, after denoising, for the 3-axis of the accelerometer and gyroscope.

4 Implemented Methods

Six widely used classifiers where implemented. In the dataset used, there are six classes, each representing a single activity. Dataset was pre-spited to training and evaluation set.

The six classifiers were chosen based on their broad usage on activity recognition field. The implementation was based on already published papers, thus the features used for training were the ones proposed in the literature.

4.1 k-Nearest Neighbours

K-Nearest Neighbors (kNN) algorithm has been extensively used on human activity recognition domain, since naturally it can better handle multi-class problems. The algorithm was tested with the number of neighbors varying from 2 to 30. Best

results were observed when the number of them was set to 18. All neighbors were equally weighted.

4.2 Decision Trees / Random Forests

Decision trees were implemented based on evaluation of different parameters, while using the features proposed on [7]. After evaluating the performance of both the Gini and entropy criterion, the former was chosen as it was yielding better results. The Random Forests ensemble method was implemented by using 70 Decision Trees trained with the previously chosen parameters.

4.3 Multilayer Perceptron

The Multilayer Perceptron (MLP) was implemented using Keras with Tensorflow backend. The MLP consists of three hidden layers, the first and second layer having 1000 neurons while the third 128. The activation function of all layers was the ReLU function and the dropout was set to 20%. The activation function for the output layer was the Softmax function. Table 1 contains the configuration and parameters used for the MLP.

Parameter	Value
Learning rate	0.005
Loss function	Mean squared error
Batch size	64
Epochs	400
Patience	40

Table 1. MLP Training parameters

4.4 Convolution Neural Networks

Convolutional Neural Network, was implemented as proposed on [10]. The structure of the CNN was a convolution layer followed by a max-pooling layer, repeated three times. ReLU function was chosen as activation function. A fully connected layer with 1000 neurons was the last layer before the output layer. The parameters can be seen on Table 2.

Parameter	Value	
Convolution layer	Feature maps	96 (192 after the 1st)
	Filter size	9
Max pooling layer	Strides	3
	Pool size	3
dropout	80%	
Loss function	Mean squared error	
Batch size	128	
Epochs	5000	
Patience	100	

Table 2. CNN training parameters

4.5 Recurrent Neural Networks

Recurrent Neural Networks, unlike feedforward networks, use an internal memory when processing inputs. A variant of RNN, the LSTM Neural Network was implemented as proposed on [11]. Table 3 contains the parameters used on the model. According to the literature, LSTM networks usually outperform vanilla RNN, as they are not affected that much by the vanishing gradient problem. In total three layers were used with 60 units each and 50% dropout, while the Softmax was chosen for activation function.

Parameter	Value
Learning rate	0.001
Loss function	Cross entropy
Batch size	20
Epochs	80
Patience	40

Table 3. RNN training parameters

4.6 Support Vector Machine

A Support Vector Machine (SVM) was also implemented. Both a linear and a gaussian kernel were tested. Since the activity recognition problem is a multiclass classification problem, the one against all technique was employed. Error tolerance was set to $e-4$ while the maximum number of iterations to 4000.

5 Results

All classifiers, were trained on the same set of activities using a common dataset. All methods, except the RNN, were trained on a machine equipped with i7-4700MQ processor 8 GB RAM and NVidia GT 745M 2GB GPU running windows. The RNN was trained and evaluated on a PC equipped with an i7-7700HQ CPU, 16GB RAM and NVidia GTX 1050M 4GB GPU, due to the increased computational power needed.
As already mentioned, the dataset was already split to training and evaluation subsets. The training set was randomly split with 70% of the subjects forming the train set and 30% the validation set. All subjects performed the same set of activity drills. After training using the training set, each classifier was evaluated against the test set. F1-score was chosen as the evaluation metric. Table 4 presents the results of the analysis for the 7 classifiers on the six given activities. Numbers in bold indicate the highest F1-score for each activity and classifier. It is obvious from Table 4 that Deep Learning techniques achieve a higher score. More importantly, CNN had the best performance with an F1-score of 0.94. It is worth mentioning that CNN as well as RNN were the only methods using raw signals as input and not extracted features from accelerometer and gyroscope data.

Classifier / Activity	KNN	DT	RF	MLP	CNN	RNN	SVM
Standing	0.67	0.62	0.72	0.75	0.97	0.96	0.75
Sitting	0.57	0.59	0.69	0.76	0.97	0.96	0.73
Laying	0.82	0.71	0.82	0.81	1.00	0.94	0.73
Walking	0.55	0.46	0.56	0.06	0.89	0.61	0.28
Walking downstairs	0.70	0.58	0.70	0.66	0.91	0.80	0.68
Walking upstairs	0.82	0.75	0.84	0.73	0.92	0.77	0.79
Total	0.69	0.62	0.72	0.62	0.94	0.84	0.68

Table 4. F1 score for each method and activity

Examining the confusion matrix (Table 5), useful results can be drawn regarding activities that have a similar accelerometer and gyroscope profile. Each row of the confusion matrix represents one of the activities performed (ground truth) and each column one of the recognized activities. Individual cells contain the number of examples that belong to class shown at the corresponding row, classified as instances of the class shown on the column. Sitting/standing, walking/walking downstairs are the pairs that were usually misclassified as each other. More specifically, out of 491 examples belonging to the walking class, the MLP had the worst performance classifying correctly only 16 examples while 289 examples were misclassified as walking downstairs. On the contrary, the Convolution Neural Network correctly assigned 446 examples to the walking class.

Regarding the sitting/standing pair of activities, the kNN and Decision Trees had the worst performance. The former classified 248 out of 471 examples correctly as sitting and 214 wrongly as standing. The latter, correctly assigned 275 examples to the sitting class and 156 to standing class. CNN and RNN had the best performance when classifying instances of the sitting class with 467 and 469 correctly assigned examples respectively. The F1-score of each method on each individual class can be seen on Table 4.

	Classifier	**Standing**	**Sitting**	**Laying**	**Walking**	**Walking downstairs**	**Walking upstairs**
Standing	KNN	387	87	22	0	0	0
	DT	323	110	63	0	0	0
	RF	373	88	35	0	0	0
	MLP	343	103	49	0	0	0
	CNN	469	8	19	0	0	0
	RNN	467	9	20	0	0	0
	SVM	410	63	23	0	0	0
Sitting	KNN	214	248	9	0	0	0
	DT	156	275	40	0	0	0
	RF	111	341	19	0	0	0
	MLP	56	391	24	0	0	0
	CNN	3	467	1	0	0	0
	RNN	1	469	1	0	0	0
	SVM	131	334	2	0	0	4
Laying	KNN	54	55	311	0	0	0
	DT	55	74	291	0	0	0
	RF	41	59	320	0	0	0
	MLP	19	61	340	0	0	0
	CNN	0	5	415	0	0	0
	RNN	2	22	396	0	0	0
	SVM	44	44	332	0	0	0
Walking	KNN	1	2	0	249	207	32
	DT	0	0	0	220	189	82
	RF	0	0	0	245	194	52
	MLP	0	1	0	16	289	185
	CNN	0	1	0	446	31	13
	RNN	0	0	0	230	117	144
	SVM	1	1	0	89	248	152
	KNN	0	0	0	92	419	21

	Classifier	Standing	Sitting	Laying	Walking	Walking downstairs	Walking upstairs
Walking downstairs	DT	1	0	0	158	306	67
	RF	0	0	0	98	399	35
	MLP	0	0	0	9	438	85
	CNN	1	0	1	21	502	7
	RNN	0	0	0	11	474	47
	SVM	0	0	0	40	424	68
Walking upstairs	KNN	2	0	0	77	45	413
	DT	0	0	0	109	32	396
	RF	0	0	0	55	21	461
	MLP	0	1	1	0	69	466
	CNN	0	0	0	43	39	455
	RNN	0	0	0	16	59	462
	SVM	0	0	0	10	30	497

Table 5. Confusion Matrix showing the number of examples assigned on each class (columns) per classifier, while performing a specific activity (row).

5.1 Activity recognition Android application

The classifier with the best performance (CNN) was chosen for evaluation on real life semi-supervised environment. The application developed was a simple app that gathered the data from the phones sensors (accelerometer & gyroscope), applied denoise filters and reported the classification results to the user. A sliding window was also used with the same parameters as the one employed on the dataset. The phone used for evaluation was a Samsung Galaxy S6. The model was exported and implemented on the phone using the libraries that Tensorflow provides.

Results from the mobile application, although promising, were not on par with results obtained from laboratory evaluation. The main reason for that is that the trained data were obtained from a 2G accelerometer and gyroscope, while modern mobile phones are equipped with a 4G, having higher sensitivity and returning values in a bigger range. In order to overcome that problem, data were normalized in the range the 2G accelerometer is working.

The android application was evaluated using 10 people between 23-35 years old in a semi-supervised environment. All subjects had the phone attached on their waist using a belt. No person specific calibration was made. All subjects performed the same set of activities for the same duration. Experiments were observed and the application was constantly reporting the recognized activity, both on screen and vocally. The activities performed were (in order): standing for approximately 30 seconds, walking approximately 40 meters, going a staircase down/up, walk approximately 10 meters, laying for 30 seconds, sitting for 30 seconds.

Results, as can be seen on Table 6, show that the Neural Network was able to correctly classify most of the performed activities. The accuracy of the CNN when used on the mobile phone was 73.87%. Due to the 50% overlap of the sliding window used, the transition between activities resulted in misclassifications, especially for activities that have a similar acceleration profile. This was a result of remnant data from the previous activity. Walking downstairs had the worst performance as it was not recognized at all. Walking upstairs also had the lowest F1 score as it was mostly misclassified as walking.

Walking activity had a low F1 score compared with the results obtained during evaluation. We consider that the main reason for that is that no user specific calibration was made. Additionally, the mobile phone had shifted from its original orientation during walking, resulting in inconsistent readings.

	Standing	Sitting	Laying	Walking	Upstairs	Downstairs
Recall	58.442%	96.154%	100%	64.122%	100%	No data
Precision	90%	100%	96%	84%	6.667%	0%
F1	70.8%	98.03%	97.9%	72.73%	12.5%	No data

Table 6. Results from smartphone application testing

6 Conclusion / Future Work

Human activity recognition using mobile devices is a research area with increased interest. Direct comparison between different methods, is not an easy task, as datasets gathered from different people and sensors are used, as well as different activities are recognized. In our work, state of the art classifiers already used in the literature were evaluated on the same set of activities, using the same set of data. Deep learning techniques, exploiting CNN and RNN methods outperformed the rest of the implemented classifiers. Confusion matrices generated, allowed the identification of activities pairs that were misinterpreted as each other. Those pairs are activities that have a similar acceleration profile.

Experiments were also conducted on real life in a semi-supervised environment. An android application was developed and the CNN (the classifier with the best overall

performance) was used to identify activities performed by participants on the experiments. Data processing and recognition was offline, i.e. on the mobile device. While further evaluation is needed, results were promising. Orientation of the device still remains a major factor affecting performance as well as the exact position of the phone and how discrete a person's movement is. The power demands of the application was also measured, in order to evaluate whether it is feasible, energy wise, to have all the preprocessing and classification on the phone. Power consumption was found to be relatively low.

In addition to the above, a computation load profiling on several CPUs is required. As computational power on mobile devices is limited, a detailed analysis is important. A profiling on RNN running on mobile phones was presented on [16].

The need for a common evaluation framework was identified during our work, a task that has to be addressed in near future. Additionally, real life testing has to be more extensive. The rest of the classifiers should also be evaluated on a smartphone. It is also expected in near future to evaluate the application on different age groups and perform the recognized activities on multiple terrains, such as roads, gravel, uphill etc.

Acknowledgements

This work has been funded by the ΕΣΠΑ (2014-2020) Erevno - Dimiourgo – Kainotomo 2018/EPAnEK Program 'Energy Controlling Voice Enabled Intelligent Smart Home Ecosystem', General Secretariat for Research and Technology, Ministry of Education, Research and Religious Affairs.

References

[1] P. Kumari, L. Mathew and P. Syal (2017). Increasing trend of wearables and multimodal interface for human activity monitoring: A review. Biosensors and Bioelectronics, (90), 298–307.

[2] O. Lara and M. Labrador (2013). A Survey on Human Activity Recognition using Wearable Sensors. In: IEEE Communications Surveys & Tutorials 3(15), 1192–1209.

[3] W. Sansrimahachai and M. Toahchoodee (2017). Mobile phone based immobility tracking system for elderly care. In: IEEE Annual International Conference Proceedings/TENCON, 3550–3553.

[4] M. Álvarez de la Concepción, L. Soria Morillo, J. Álvarez García, L. González-Abril (2017). Mobile activity recognition and fall detection system for elderly people using Ameva algorithm. In: Pervasive and Mobile Computing (34), 3–13.

[5] N. A. Capela, E. D. Lemaire, N. Baddour, M.Rudolf, N. Goljar and H. Burger (2016). Evaluation of a smartphone human activity recognition application with able-bodied and stroke participants. In: Journal of NeuroEngineering and Rehabilitation, 13(1), 5.

[6] T. Sztyler, H. Stuckenschmidt and W. Petrich (2017). Position-aware activity recognition with wearable devices. In: Pervasive and Mobile Computing (38), 281–295.

[7] C. Chen and W. Lee (2017). Enabling Human Activity Recognition with Smartphone Sensors in a Mobile Environment. In: World Congress on Engineering Proceedings, vol. 1, 5–9.

[8] J. Santiago, E. Cotto, L. G. Jaimes and I. Vergara-Laurens (2017). Fall detection system for the elderly. In: IEEE 7th Annual Computing and Communication Workshop and Conference, 1-4.

[9] M. Shoaib, S. Bosch, O. Durmaz Incel, H. Scholten and P. J. M. Havinga (2016). Complex Human Activity Recognition Using Smartphone and Wrist-Worn Motion Sensors. In: Sensors (Switzerland) 16(4), 1–24.

[10] C. A. Ronao and S. B. Cho (2016). Human activity recognition with smartphone sensors using deep learning neural networks. In: Expert Systems with Applications (59), 235–244.

[11] M. Inoue, S. Inoue and T. Nishida (2018). Deep Recurrent Neural Network for Mobile Human Activity Recognition with High Throughput. In: Artificial Life and Robotics. 23(2), 173-185.

[12] I. Pires, N. Garcia, N. Pombo, F. Flórez-Revuelta, S. Spinsante and M. Teixeira (2018). Identification of Activities of Daily Living through Data Fusion on Motion and Magnetic Sensors embedded on Mobile Devices. In: Pervasive and Mobile Computing, 47, pp. 78-93.

[13] I. Pires, N. Garcia, N. Pombo, F. Flórez-Revuelta and S. Spinsante (2018). Approach for the development of a Framework for the Identification of Activities of Daily Living using Mobile Devices. In: Sensors, 18(2), pp. 640.

[14] D. Anguita, A. Ghio, L. Oneto, X. Parra and J. L. Reyes-Ortiz (2013). A Public Domain Dataset for Human Activity Recognition Using Smartphones. In: European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning, pp. 24–26.

[15] R. Chavarriaga, et al. (2013). The Opportunity challenge: A benchmark database for on-body sensor-based activity recognition. In: Pattern Recognition Letters, 15(34), 2033–2042.

[16] Q. Cao, N. Balasubramanian and A. Balasubramanian (2017). MobiRNN: Efficient reccurent neural network execution on mobile GPU. In: Proceedings of the 1st International Workshop on Deep Learning for Mobile Systems and Applications, 1-6.

.