

# Building Personalized Activity Recognition Models with Scarce Labeled Data Based on Class Similarities

Enrique Garcia-Ceja<sup>(✉)</sup> and Ramon Brena

Tecnológico de Monterrey, Campus Monterrey, Av. Eugenio Garza Sada 2501 Sur,  
Monterrey, Nuevo León, Mexico

e.g.mx@ieee.org, ramon.brena@itesm.mx

**Abstract.** With the recent advent of new devices with embedded sensors, Human Activity Recognition (HAR) has become a trending topic in the last years because of its potential applications in pervasive health care, assisted living, exercise monitoring, etc. Most of the works on HAR either require from the user to label the activities as they are performed so the system can learn them, or rely on a trained device that expects a “typical” ideal user. The first approach is impractical, as the training process easily become time consuming, expensive, etc., while the second one drops the HAR precision for many non-typical users. In this work we propose a “crowdsourcing” method for building personalized models for HAR by combining the advantages of both *user-dependent* and *general* models by finding class similarities between the target user and the community users. We evaluated our approach on 4 different public datasets and showed that the personalized models outperformed the user-dependent and general models when labeled data is scarce.

**Keywords:** Activity recognition · Wearable sensors · Accelerometer · Model personalization

## 1 Introduction

In the last years Human Activity Recognition (HAR) [1] has gained a lot of attention because of its wide range of applications in several areas such as health and elder care, sports, etc. [2–4]. Inferring the current activity being performed by an individual or group of people can provide valuable information in the process of understanding the context and situation in a given environment and as a consequence, personalized services can be delivered. Recently, the use of wearable sensors has become the most common approach to recognize physical activities because of its unobtrusiveness and ubiquity –specifically the use of accelerometers [4–6] because they are already embedded in several devices and they raise less privacy concerns than other types of sensors.

One of the problems in HAR systems is that the labeling process for the training data tends to be tedious, time consuming, difficult and prone to errors.

This problem has really hindered the practical application of HAR systems, limiting them to the most basic activities, for which a general model is enough, as is the case for the pedometer function or alerting the user who spends too much time quiet sitting down; both functions now available in some fitting devices and smartwatches.

On the other hand, when trying to offer personalized HARs, there is the problem that at the initial state of a system there is little or no information at all (in our case, sensor data and labels). In the field of *recommender systems* (e.g., movie, music, book recommenders) this is known as the *cold-start problem* [7] and it includes the situation when there is a new user but nothing or little is known about him/her, in which case it becomes difficult to recommend an item, service, etc. It also encompasses the situation when a new item is added to the system but since no one has yet rated, purchased or used that item, then it is difficult to recommend it to the users.

In this work, we will focus in the situation when there is a new user in the system and we want to infer his/her physical activities from sensor data with high accuracy even when there is little information about that particular user, assuming that the system already has data from many other users and also that their associated data is already labeled. We are thus attempting to use a “crowdsourcing” approach which consists in using collective data to fit personal data. The key insight in our approach is that instead of building a model with all the data from all other users, we will use the scarce labeled data from the target user to select a subset of the other users’ data based on class similarities to build a personalized model. The rationale behind this idea is that the way people move varies between individuals so we want to exclude instances from the training set that are very different from those of the target user in order to remove noise.

This paper is organized as follows: Sect. 2 presents some related work. Section 3 details the process of building a Personalized Model. The experiments are described in Sect. 4. Finally in Sect. 5 we draw our conclusions.

## 2 Related Work

From the reviewed literature, broadly three different types of models in HAR can be identified—namely: *General*, *User-Dependent*, and *Mixed* models.

*General Models (GM)*: Sometimes also called *User-Independent Models*, *Impersonal Models*, etc. and from now on we will refer to them as GMs. For each specific user  $i$  a model is constructed using the data from all other users  $j$ ,  $j \neq i$ ; the accuracy is calculated testing the model with the data from user  $i$ .

*User-Dependent Models (UDM)*: They are also called *User-Specific Models*, here we will refer to them as UDMs. In this case, individual models are trained and evaluated for a user using just her/his own data.

*Mixed Models (MM)*: In [8] they call them Hybrid models. This type of model tries to combine GMs and UDMs in the hope of adding their respective strengths, and usually is trained using all the aggregated data without distinguishing between users.

There are some works in HAR that have used the UDM and/or GM approach [9–11]. The disadvantages of GMs are mostly related to their lack of precision, because the data from many dissimilar users is just aggregated. This limits the GM HAR systems to very simple applications such as pedometers and detection of long periods of sitting down. The disadvantages of UDM HAR systems are related to the difficulties of labeling the specific users' data, as the training process easily become time consuming and expensive, so in practice users avoid it.

For UDMs, several techniques have been used to help users label the data, as it is the weakest link in the process. For example, in [12] a mobile application was built in which the user can select several activities from a predefined list. In [13], they first video-recorded the data collection session and then manually labeled the data. Some other works have used a Bluetooth headset combined with speech recognition software to perform the annotations [14] whereas in [15] the annotations were made manually by taking notes. Anyway, labeling personal activities remains being very time-consuming and undesirable indeed.

From the previous comments, apparently MMs look like a very promising approach, because they could cope with the disadvantages of both GM and UDM, but in practice combining the strengths of both has been an elusive goal; as noted by Lockhart & Weiss [8], no such system has made it to actual deployment.

There have been several works that have studied the problem of scarce labeled data in HAR systems [16, 17] and used Semi-supervised learning methods to deal with the problem, however they follow a *Mixed* model approach, i.e., they do not distinguish between users.

Model *personalization/adaptation* refers to training and adapting classifiers for a specific user according to his/her own needs. Building a model with data from many users and using it to classify activities for a target user will introduce noise due to the diversity between users. Lane et al. [18] showed that there is a significant difference for the *walking* activity between two different groups of people (20–40 and > 65 years old). Parviainen et al. [19] also argued that a single general model for activity classification will not perform well due to individual differences and proposed an algorithm to adapt the classification for each individual by only requesting binary feedback from the user. In [20] they used a model adaptation algorithm (Maximum A Posteriori) for stress detection using audio data. Zheng et al. [21] used a collaborative filtering approach to provide targeted recommendations about places and activities of interest based on GPS traces and annotations. They manually extracted the activities from text annotations whereas in this work the aim is to detect *physical* activities from accelerometer data. Abdallah et al. [22] proposed an incremental and active learning approach for activity recognition to adapt a classification model as new sensory data arrives. In [23] they proposed a personalization algorithm that uses clustering and a Support Vector Machine that first, trains a model using data from user A and then personalizes it for another person B, however they did not specify how should user A be chosen. This can be seen as a  $1 \rightarrow n$  relationship in the sense that the base model is built using data from a specific

user A and the personalization of all other users is based solely on A. The drawback of this approach is that user A may be very different from all other users which could lead to poor final models. Our work differs from this one in that we follow a  $n \rightarrow 1$  approach which is more desirable in real world scenarios, i.e., use data already labeled by the community users to personalize a model for a specific user. In work [18] they personalize models for each user by first building Community Similarity Networks (CSN) for different dimensions such as: physical similarity, lifestyle similarity and sensor-data similarity. Our study differs from this one in two key aspects: First, instead of looking for inter-user similarities we find similarities between classes of activities. This is because two users may be similar overall but still, there may be activities that are performed very differently between them. Second, we just use accelerometer data to find similarities since other types of data (age, locations, height, etc.) are usually not available or impose privacy concerns. Furthermore, we evaluated the proposed method on 4 different public datasets collected by independent researchers.

In this work we will use an approach that is between GMs and UDMs, so it could be seen as a variation of Mixed Models, but here we use a small amount of the user's available data to select a subset of the other users' activities instances to complement the data from the considered user, instead of just blindly aggregating all other users' data. This selection is made based on class similarities and the details will be presented in Sect. 3.

### 3 Personalized Models

In this section we describe how a Personalized Model (PM) is trained for a given target user  $u_t$ . A General Model (GM) includes all instances from users  $U_{other}$ , where  $U_{other}$  is the set of all users excluding the target user  $u_t$ . In this case there may be differences between users on how they perform each activity (e.g., some people tend to walk faster than others) so this approach will introduce noisy instances to the train set and thus, the resulting model will not be very accurate when recognizing activities for  $u_t$ .

The idea of building a PM is to use the scarce labeled data of  $u_t$  to select instances from a set of users  $U_{similar}$ , where  $U_{similar}$  is the set of users similar to  $u_t$  according to some similarity criteria. Building PMs for activity recognition was already studied by Lane et al. [18], with the limitations we already explained in the preceding section. In our approach, we look for similarities per class instead of a per user basis, i.e., the final model will be built using only the instances that are similar to those of  $u_t$  for each class. Procedure 1 presents the proposed algorithm to build a PM based on class similarities.

The procedure starts by iterating through each possible class  $c$ . Within each iteration, instances of class  $c$  from the  $u_t$  train set  $\tau_t$  and all the instances of class  $c$  that belong to all other users are stored in  $data_{all}$ . The function  $subset(set, c)$  returns all the instances in  $set$  of class  $c$  which are then saved in  $data_t$ . Function  $instances(U)$  returns all the instances that belong to the set of users  $U$ . Next, all instances in  $data_{all}$  are clustered using k-means algorithm for

**Procedure 1.** Build PM

---

```

1:  $T \leftarrow \{\}$  ▷ Start with an empty train set
2: for  $c$  in  $C$  do ▷ For each class
3:    $data_t \leftarrow subset(\tau_t, c)$  ▷  $\tau_t$  is the target user's train set
4:    $data_{other} \leftarrow subset(instances(U_{other}), c)$ 
5:    $data_{all} \leftarrow data_t \cup data_{other}$ 
6:   Cluster  $data_{all}$  using k-means for  $k = 2..UpperBound$  and select the optimal  $k$ 
   according to some clustering quality index.
7:    $S \leftarrow \arg \max_{g \in G} |data_t \cap g|$  ▷  $G$  is the set of the resulting  $k$  groups
8:    $T \leftarrow T \cup S \cup data_t$ 
9: end for
10:  $weight(T)$  ▷ Assign a weight to each instance such that the importance of  $\tau_t$ 
   increases as more training data of the target user is available.
11: Build model using training set  $T$ .

```

---

$k = 2..UpperBound$ . For each  $k$ , the *Silhouette* clustering quality index [24] of the resulting groups is computed and the  $k$  that produces the optimal quality index is chosen. A clustering quality index [25] is a measure of the *quality* of the resulting clustering based on compactness and separation. The *Silhouette* index was chosen because it has been shown to produce good results with different datasets [25]. Next, instances from the cluster in which the majority of instances from  $data_t$  ended up are added to the final training set  $T$ . Also all instances from  $data_t$  that ended up in other clusters are added to  $T$  to make sure all the data from  $u_t$  are used. After the *for* loop, all instances in  $T$  are assigned an *importance* weight as a function of the size of  $\tau_t$  such that instances from the  $u_t$  train set have more impact as more training data is available for that specific user. The exponential decay function  $y = (1 - r)^x$  was used to assign the weights where  $r$  is a decay rate parameter and  $x = |\tau_t|$ . The weight of all instances in  $T$  that are not in  $\tau_t$  is set to  $y$  and the weight of all instances in  $\tau_t$  is set to  $1 - y$ . Finally, the model is built using  $T$  with the new instances' weights. Note that the classification model needs to have support for instance weighting. In this case we used a decision tree implementation called *rpart* [26], which supports instance weighting.

## 4 Experiments and Results

We conducted our experiments with 4 publicly available datasets. *D1: Chest Sensor Dataset* [27, 28]; *D2: Wrist Sensor Dataset* [29, 30]; *D3: WISDM Dataset* [31, 32]; *D4: Smartphone Dataset* [13, 33]. For datasets D1 and D2, 16 common statistical features on fixed length windows were extracted. The features were: mean for each axis, standard deviation for each axis, maximum value of each axis, correlation between each pair of axes, mean of the magnitude, standard deviation of the magnitude, mean difference of the magnitude, and area under the curve of the magnitude. D3 already included 46 features and D4 already included 561 extracted features from the accelerometer and gyroscope sensors.

Several works in HAR perform the experiments by first collecting data from one or several users and then evaluating their methods using *k-fold cross validation* (being 10 the most typical value for *k*) on the aggregated data. For a *k* = 10 this means that all the data is randomly divided into 10 subsets of approximately equal size. Then, 10 iterations are performed. In each iteration a subset is chosen as the test set and the remaining *k* - 1 subsets are used as the train set. This means that 90% of the data is completely labeled and the remaining 10% is unknown, however, in real life situations it is more likely that just a fraction of the data will be labeled. In our experiments we want to consider the situation when the target user has just a small amount of labeled data. Our models' evaluation procedure consists of sampling a small percent *p* of instances from *u<sub>t</sub>* to be used as the train set  $\tau_t$  and use the remaining data to test the performance of the General Model, User-Dependent Model and our proposed Personalized Model. To reduce sampling variability of the train set we used proportionate allocation stratified sampling. We chose *p* to range between 1% to 30% with increments of 1. For each *p* percent we performed 5 random sampling iterations for each user.

Figures 1, 2, 3 and 4 show the results of averaging the accuracy of all users for each *p* percent of data used as train set. For D1 (Fig. 1) the PM clearly outperforms the other two models when the labeled data is between 1% and 10% (the curve for PM-2 will be explained soon). The GM shows a stable accuracy since it is independent of the user. For the rest of the datasets the PM shows an overall higher accuracy except for D2 (later we will analyze why this happened).

Table 1 shows the average number of labeled instances per class for each *p* percent of training data. For example for D3 we can see how just using 3 labeled instances per class the PM achieves a good classification accuracy ( $\approx 0.8$ ).

Table 2 shows the difference of average overall accuracy and recall (from 1% to 30% of labeled data) between the PM and the other two models. Here we can

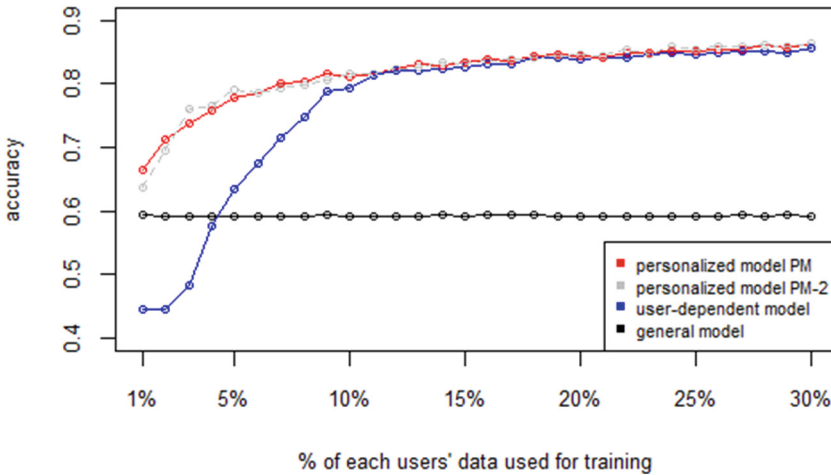


Fig. 1. D1: Chest sensor dataset

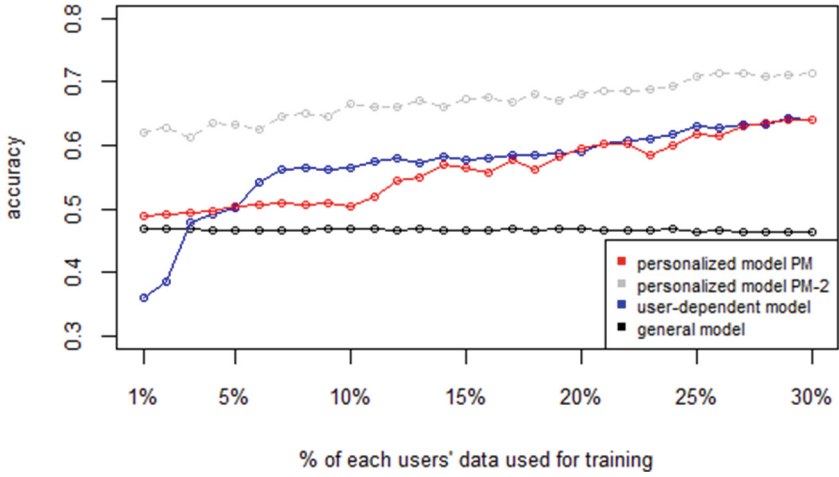


Fig. 2. D2: Wrist sensor dataset

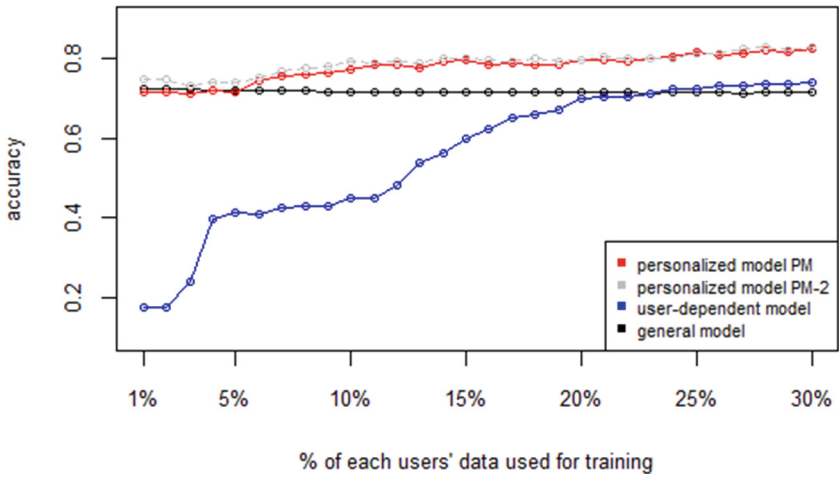


Fig. 3. D3: WISDM dataset

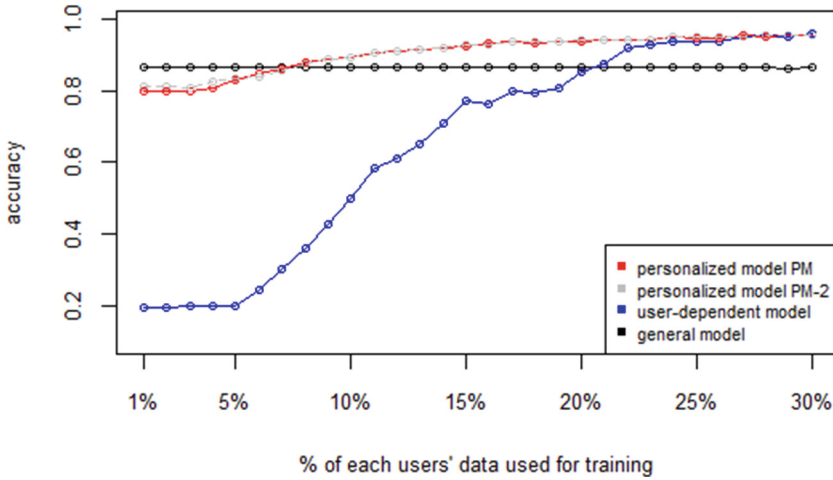
see how the PM significantly outperforms the other two models in all datasets except for the accuracy in D2 when comparing PM - UDM in which case the difference is negligible. This may be due to the user-class sparsity of the dataset, i.e., some users just performed a small subset of the activities. This situation will introduce noise to the PM. In the extreme case when a user has just 1 type of activity it would be sufficient to always predict that activity. However, the PM is trained with the entire set of possible labels from all other users in which case the model will predict labels that are not part of that user. To confirm this, we visualized and quantified the user-class sparsity of the datasets and performed

**Table 1.** Average number of labeled instances per class.

	1 %	5 %	10 %	15 %	20 %
D1	1	7	14	21	28
D2	1	1	2	3	3
D3	1	2	3	4	5
D4	1	3	6	9	12

**Table 2.** Difference of average overall accuracy/recall (from 1 % to 30 % of labeled data) between the PM and the other two models.

	PM - GM	PM - UDM
D1	22.5 %/18.4 %	5 %/8 %
D2	9.3 %/9.4 %	-0.80 %/17 %
D3	6 %/7 %	22.15 %/34 %
D4	4 %/4 %	25.5 %/28 %



**Fig. 4.** D4: Smartphone dataset

further experiments. First we computed the user-class sparsity matrices for each dataset. These matrices are generated by plotting what activities were performed by each user. A cell in the matrix is set to 1 if a user performed an activity and 0 otherwise. The sparsity index is computed as 1 minus the proportion of 1's in the matrix. In datasets D1 and D4 all users performed all activities giving a sparsity index of 0. Figures 5 and 6 show the user-class sparsity matrices of datasets D2 and D3 respectively. D2 has a sparsity index of 0.54 whereas for D3 it is 0.18. For D2 this index is very high (almost half of the entries in the matrix are 0) furthermore the number of classes for this dataset is also high (12). From the matrix we can see that several users performed just a small number of activities (in some cases just 1 or 2 activities). One way to deal with this situation is to train the model excluding activities from other users that were not performed by the target user. Figures 1, 2, 3 and 4 (gray dotted line PM-2) show the results of excluding types of activities that are not in  $u_t$ . As expected, for datasets with low or no sparsity the results are almost the same (with small



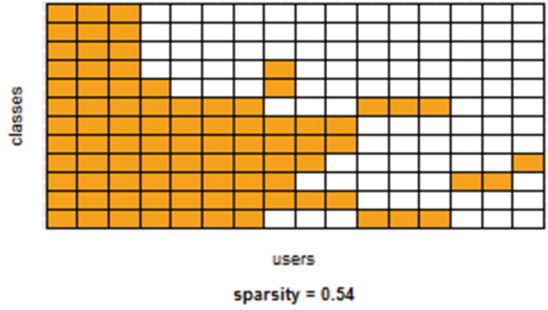


Fig. 5. D2: wrist sensor dataset user-class sparsity matrix

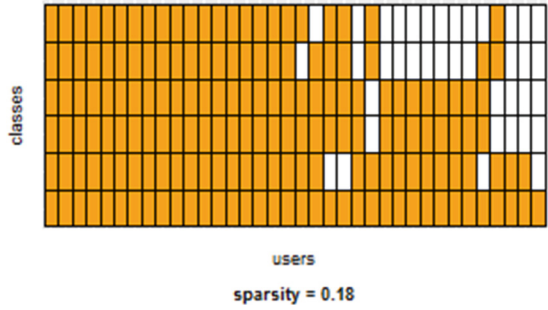


Fig. 6. D3: WISDM dataset user-class sparsity matrix

variations due to random initial k-means centroids). For D2 which has a high sparsity the accuracy significantly increased. This shows evidence that the user-class distribution of the dataset has an impact on the PM and that this can be alleviated by excluding the classes that are not relevant for a particular user.

## 5 Conclusions

In this work we proposed a method based on class similarities between a collection of previous users and a specific user to build Personalized Models when labeled data for this one is scarce, getting thus the benefits from a “crowdsourcing” approach, where the community data is fit to the individual case. We used the small amount of labeled data from the specific user to select meaningful instances from all other users in order to reduce noise due to inter-user diversity. We evaluated the proposed method on 4 independent human activity datasets. The results showed a significant increase in accuracy over the General and User-Dependent Models for datasets with small sparsity. In the case of datasets with high sparsity, the performance problems were alleviated in a great extent by excluding types of activities from other users that were not performed by the target user.

**Acknowledgements.** Enrique Garcia-Ceja would like to thank Consejo Nacional de Ciencia y Tecnología (CONACYT) and the AAAMI research group at Tecnológico de Monterrey for the financial support in his PhD. studies.

## References

1. Brush, A., Krumm, J., Scott, J.: Activity recognition research: the good, the bad, and the future. In: *Proceedings of the Pervasive 2010 Workshop on How to Do Good Research in Activity Recognition*, Helsinki, Finland, pp. 17–20 (2010)
2. Martínez-Pérez, F.E., González-Fraga, J.Á., Cuevas-Tello, J.C., Rodríguez, M.D.: Activity inference for ambient intelligence through handling artifacts in a health-care environment. *Sensors* **12**(1), 1072–1099 (2012)
3. Han, Y., Han, M., Lee, S., Sarkar, A.M.J., Lee, Y.-K.: A framework for supervising lifestyle diseases using long-term activity monitoring. *Sensors* **12**(5), 5363–5379 (2012)
4. Mitchell, E., Monaghan, D., O’Connor, N.E.: Classification of sporting activities using smartphone accelerometers. *Sensors* **13**(4), 5317–5337 (2013)
5. Banos, O., Galvez, J.-M., Damas, M., Pomares, H., Rojas, I.: Window size impact in human activity recognition. *Sensors* **14**(4), 6474–6499 (2014)
6. Andrea Mannini and Angelo Maria Sabatini: Machine learning methods for classifying human physical activity from on-body accelerometers. *Sensors* **10**(2), 1154–1175 (2010)
7. Schein, A.I., Popescul, A., Ungar, L.H., Pennock, D.M.: Methods and metrics for cold-start recommendations. In: *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 253–260. ACM (2002)
8. Lockhart, J.W., Weiss, G.M.: Limitations with activity recognition methodology & data sets. In: *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication, UbiComp 2014 Adjunct*, pp. 747–756. ACM, New York (2014)
9. Varkey, J.P., Pompili, D., Walls, T.A.: Human motion recognition using a wireless sensor-based wearable system. *Pers. Ubiquit. Comput.* **16**(7), 897–910 (2012)
10. Khan, A.M., Lee, Y.-K., Lee, S.Y., Kim, T.-S.: A triaxial accelerometer-based physical-activity recognition via augmented-signal features and a hierarchical recognizer. *IEEE Trans. Inf. Technol. Biomed.* **14**(5), 1166–1172 (2010)
11. Zhang, M., Sawchuk, A.A.: A feature selection-based framework for human activity recognition using wearable multimodal sensors. In: *Proceedings of the 6th International Conference on Body Area Networks, ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering)*, pp. 92–98 (2011)
12. Lara, Ó.D., Pérez, A.J., Labrador, M.A., Posada, J.D.: Centinela: a human activity recognition system based on acceleration and vital sign data. *Pervasive Mob. Comput.* **8**(5), 717–729 (2012)
13. Anguita, D., Ghio, A., Oneto, L., Parra, X., Reyes-Ortiz, J.L.: Human activity recognition on smartphones using a multiclass hardware-friendly support vector machine. In: Bravo, J., Hervás, R., Rodríguez, M. (eds.) *IWAAL 2012. LNCS*, vol. 7657, pp. 216–223. Springer, Heidelberg (2012)
14. Khan, A.M., Lee, Y.-K., Lee, S., Kim, T.-S.: Accelerometers position independent physical activity recognition system for long-term activity monitoring in the elderly. *Med. Biol. Eng. Comput.* **48**(12), 1271–1279 (2010)

15. Garcia-Ceja, E., Brena, R.F., Carrasco-Jimenez, J.C., Garrido, L.: Long-term activity recognition from wristwatch accelerometer data. *Sensors* **14**(12), 22500–22524 (2014)
16. Guan, D., Yuan, W., Lee, Y.-K., Gavrilov, A., Lee, S.: Activity recognition based on semi-supervised learning. In: 13th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications, 2007, RTCSA 2007, pp. 469–475 (2007)
17. Stikic, M., Van Laerhoven, K., Schiele, B.: Exploring semi-supervised and active learning for activity recognition. In: 12th IEEE International Symposium on Wearable Computers, 2008, ISWC 2008, pp. 81–88. IEEE (2008)
18. Lane, N.D., Xu, Y., Lu, H., Hu, S., Choudhury, T., Campbell, A.T., Zhao, F.: Enabling large-scale human activity inference on smartphones using community similarity networks (Csn). In: Proceedings of the 13th International Conference on Ubiquitous Computing, UbiComp 2011, pp. 355–364. ACM, New York (2011)
19. Parviainen, J., Bojja, J., Collin, J., Leppänen, J., Eronen, A.: Adaptive activity and environment recognition for mobile phones. *Sensors* **14**(11), 20753–20778 (2014)
20. Lu, H., Frauendorfer, D., Rabbi, M., Mast, M.S., Chittaranjan, G.T., Campbell, A.T., Gatica-Perez, D., Choudhury, T.: StressSense: detecting stress in unconstrained acoustic environments using smartphones. In: Proceedings of the 2012 ACM Conference on Ubiquitous Computing, UbiComp 2012, pp. 351–360. ACM, New York (2012)
21. Zheng, V.W., Cao, B., Zheng, Y., Xie, X., Yang, Q.: Collaborative filtering meets mobile recommendation: a user-centered approach. In: AAAI, vol. 10, pp. 236–241 (2010)
22. Abdallah, Z.S., Gaber, M.M., Srinivasan, B., Krishnaswamy, S.: StreamAR: incremental and active learning with evolving sensory data for activity recognition. In: 2012 IEEE 24th International Conference on Tools with Artificial Intelligence (ICTAI), vol. 1, pp. 1163–1170 (2012)
23. Vo, Q.V., Hoang, M.T., Choi, D.: Personalization in mobile activity recognition system using K-medoids clustering algorithm. *Int. J. Distrib. Sens. Netw.* **2013**(315841), 12 (2013). doi:[10.1155/2013/315841](https://doi.org/10.1155/2013/315841)
24. Rousseeuw, P.J.: Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *J. Comput. Appl. Math.* **20**, 53–65 (1987)
25. Arbelaitz, O., Gurrutxaga, I., Muguerza, J., Pérez, J.M., Perona, I.: An extensive comparative study of cluster validity indices. *Pattern Recogn.* **46**(1), 243–256 (2013)
26. Therneau, T.M., Atkinson, E.J.: An introduction to recursive partitioning using the rpart routines. Technical report 61 (1997)
27. Casale, P., Pujol, O., Radeva, P.: Personalization and user verification in wearable systems using biometric walking patterns. *Pers. Ubiquit. Comput.* **16**(5), 563–580 (2012)
28. Activity recognition from single chest-mounted accelerometer data set (2012). <https://archive.ics.uci.edu/ml/datasets/Activity+Recognition+from+Single+Chest-Mounted+Accelerometer>. Accessed 2015
29. Bruno, B., Mastrogiovanni, F., Sgorbissa, A.: A public domain dataset for adl recognition using wrist-placed accelerometers. In: 2014 RO-MAN: The 23rd IEEE International Symposium on Robot and Human Interactive Communication, pp. 738–743 (2014)
30. Dataset for adl recognition with wrist-worn accelerometer data set (2014). <https://archive.ics.uci.edu/ml/datasets/Dataset+for+ADL+Recognition+with+Wrist-worn+Accelerometer>. Accessed 2015

31. Kwapisz, J.R., Weiss, G.M., Moore, S.A.: Activity recognition using cell phone accelerometers. SIGKDD Explor. Newsl. **12**(2), 74–82 (2011)
32. Activity prediction dataset (2012) . <http://www.cis.fordham.edu/wisdm/dataset.php>. Accessed 2015
33. Human activity recognition using smartphones data set (2012). <http://archive.ics.uci.edu/ml/datasets/Human+Activity+Recognition+Using+Smartphones>. Accessed 2015