A Study on Hyperparameter Configuration for Human Activity Recognition

Kemilly D. Garcia[1,3(✉)], Tiago Carvalho[2], João Mendes-Moreira[2], João M. P. Cardoso[2], and André C. P. L. F. de Carvalho[3]

[1] EWI-DMB, University of Twente, Enschede, The Netherlands
k.dearogarcia@utwente.nl
[2] INESC TEC, Faculty of Engineering, University of Porto, Porto, Portugal
{t.carvalho,jmoreira,jmpc}@fe.up.pt
[3] ICMC, University of São Paulo, São Carlos, SP, Brazil
andre@icmc.usp.br

Abstract. Human Activity Recognition is a machine learning task for the classification of human physical activities. Applications for that task have been extensively researched in recent literature, specially due to the benefits of improving quality of life. Since wearable technologies and smartphones have become more ubiquitous, a large amount of information about a person's life has become available. However, since each person has a unique way of performing physical activities, a Human Activity Recognition system needs to be adapted to the characteristics of a person in order to maintain or improve accuracy. Additionally, when smartphones devices are used to collect data, it is necessary to manage its limited resources, so the system can efficiently work for long periods of time. In this paper, we present a semi-supervised ensemble algorithm and an extensive study of the influence of hyperparameter configuration in classification accuracy. We also investigate how the classification accuracy is affected by the person and the activities performed. Experimental results show that it is possible to maintain classification accuracy by adjusting hyperparameters, like window size and window overlap, depending on the person and activity performed. These results motivate the development of a system able to automatically adapt hyperparameter settings for the activity performed by each person.

Keywords: Human Activity Recognition · Ensemble of classifiers · Semi-supervised learning · Mobile computing

This work was done in the context of the CONTEXTWA project and was partially funded by the ERDF - European Regional Development Fund through the Operational Programme for Competitiveness and Internationalisation - COMPETE 2020 Programme and by National Funds through the Portuguese funding agency, FCT - Fundação para a Ciência e a Tecnologia within project POCI-01-0145-FEDER-016883.

F. Martínez Álvarez et al. (Eds.): SOCO 2019, AISC 950, pp. 47–56, 2020.
https://doi.org/10.1007/978-3-030-20055-8_5

1 Introduction

Advanced mobile devices, such as smartphones, are usually integrated with several sensors capable of any-time sensing and data collection. The different types of motions sensors, such as accelerometers, gyroscopes and magnetometers, allow mobile devices to obtain substantial user-related information by monitoring and tracking movements of their users [3].

Human Activity Recognition (HAR) is a machine learning task focused on the use of sensing technologies to classify human activities and to infer human behavior [1]. Extensive research has been carried out in this area in the last decade [4–7], for applications like health and well-being [2], mobile security [3,9] and elderly care [1].

Most approaches of HAR found in the literature are based on supervised learning algorithms and assume that the data true label is always available. However, this assumption may not be feasible in real online scenarios, when labeled data is rare and the system feedback has to occur at runtime. As an example, in a fall detection system for elderly care, the classification feedback must occur as close as possible to the real moment of the user's fall [1].

Besides, as human beings perform activities differently, dissonant input signals are expected for the same activity [8]. To keep accuracy over time, classification models, used in HAR systems, need to be adapted to the current user. However, due to limitations of most mobile devices, different hardware resources need to be manage, such as battery and execution power, in order to keep the system efficiently working and accurate over time. Thus, there is a trade-off between amount of processed information and the resources available.

This work is based on an ensemble classifier firstly described in [15]. This algorithm has two phases, an offline and an online phase. In the beginning, the offline phase, the ensemble model is trained with labeled data from several users. In the online phase, this ensemble is used as a basic model to classify activities from a specific user, not present in the ensemble training. The ensemble model can be updated online with the user's data, if the classification has a high confidence factor.

The main contributions of this work are the extensive study of two hyperparameters important for HAR classification: the window size and the overlapping between windows (overlap factor). We analyze the impact of these hyperparameters in the model classification accuracy. Additionally, we conducted experiments with an ODROID-XU+E board[1] to evaluate the impact of these hyperparameters regarding energy consumption and execution time in a hardware similar to a smartphone.

This paper is structured as follows. Section 2 presents the related work on HAR and window parameterization. In Sect. 3 we describe the methodology

[1] ODROID-XU+E is a board mainly consisting of an Exynos5 Octa SoC, which includes 2 quad cores ARM CPUs and a PowerVR GPU, and a power measurement circuit to measure CPU, GPU and DRAM power consumption. The Exynos5 Octa SoC has been used in a number of families of smartphones.

applied in this study. The results obtained with the experiments are presented and discussed in Sect. 4. Finally, in Sect. 5, we summarize our main conclusions and point out future work directions.

2 Related Work

Dobbins et al. [2] propose an approach that uses personal data to better infer lifestyle choices for its users. Considering only labeled data, they evaluate the predictive performance of 10 supervised HAR classifiers in terms of accuracy and mobile system performance (execution time and energy consumption). Their experimental setup is based on a fixed window size of 512 samples and overlap factor of 0.5, i.e., 256 samples are reused from the previous window and only 256 new samples are used for the current window. They suggest that the sensing data should be processed in the cloud and not in the device. However, personal privacy and Internet connection are not considered. Furthermore, all data used is labeled, which cannot be guaranteed in a real online mobile system. The datasets used in the experiments contain complex activities and different user's data, but the results are not compared in terms of accuracy per user.

Mannini et al. [10] propose an SVM classifier to detect 4 activities from 33 different users. The classifier performance was tested for different window sizes, but not for the overlap between consecutive windows. Also, they do not compare, in terms of execution time, the classification task with different window sizes. The results show large variability among users performing the same activity, due to the problem of different sensor body location.

Window size has also been discussed by other authors. For example, [11–14] compare the predictive performance of classifiers over a set of window sizes. However, most of the studies do not consider the use of overlap factor and the impact of the user on the obtained accuracy.

In [11] it is presented an extensive review of the literature in window size and HAR. The accuracy of several classifiers was analyzed for different window sizes, but not regarding users. Additionally, the experimental setup was not elaborated with a leave-one-user-out, which would be a more realist approach. Instead, they used a cross validation approach, which is more affected by user variability than leave-one-participant-out.

The study conducted in this paper uses the PAMAP2 public dataset [15]. PAMAP2 includes a vast number of sensors and more complex activities than the data used by many of others studies. This dataset allows the study of the impact on HAR accuracy for different window sizes, users and activities.

3 Activity Recognition Overview

The HAR classification task can be split into 4 main steps, as illustrated in Fig. 1. The steps 1, 2 and 3 are the training phase with multiples users. The steps 1, 2 and 4 are used for the online user-specific classification.

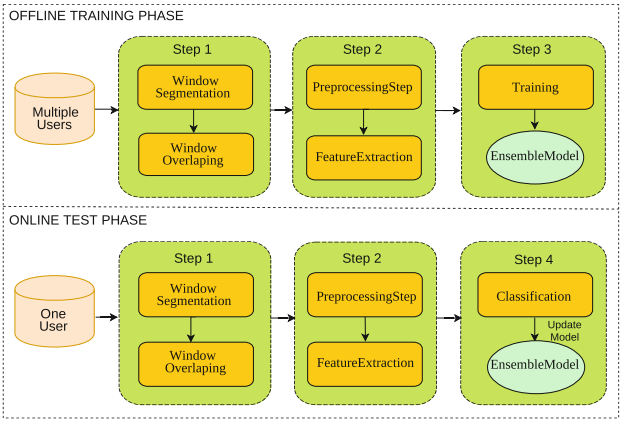


Fig. 1. Overview of the semi-supervised ensemble model for HAR.

Figure 1 shows a batch with raw data extracted from different wearable sensors and/or smartphones. The raw data samples are stored in a sliding window with fixed size. Ideally, a sliding window should contain data from a unique activity. However perfect segmentation is not always feasible, so a between-window overlap factor can be used to include samples from sequential activities. Also the size of the sliding window is reduced by the overlap factor, allowing a reduction of stored data. Thus, the step 1 is the window segmentation of the raw data and the overlapping of sequential windows.

Sensor's data are usually susceptible to noise, especially the accelerometers data [2]. Thus, it is important to process and convert the data into meaningful values. A pre-processing step (step 2) may also include calibration and filtering of the input signals in order to reduce noise. Sequential to that, a Feature Extraction (step 2) is used to calculate a single instance containing features that are then used for building the ensemble model. These features (see, e.g., [16]) include time-domain calculus, specifically mean and standard deviation for each sensor signal and correlation (Pearson correlation) between axes for the 3D sensors.

Each new instance is used to train (step 3) an ensemble model composed by three classifiers: kNN, VFDT and Naive Bayes. The implementation of the ensemble classifier is the combination of Democratic Co-Learning [17] and Tri-Training [18].

After training the ensemble model, in the online phase, sensors data are acquired from a single user. This data is pre-processed and features are extracted from them, similar to the processes (step 1 and 2) described in training phase. Each generated instance is classified by the ensemble (step 4), which classifies the instance and provides a confidence factor for that classification. The instances classified with high confidence, more than 99% value, are used to update the ensemble model.

4 Experimental Results

We conducted several experiments with our approach using the PAMAP2 dataset [15]. The objectives of these experiments are: compare the accuracy of a supervised HAR versus a semi-supervised HAR when using different configurations of the hyperparameters: window size and overlap factor. We also intend to study a HAR system behavior with the different hyperparameters configurations in terms of classification accuracy, energy consumption and execution time.

4.1 The PAMAP2 Dataset

The PAMAP2 [15] is a public dataset for human physical activities[2]. The data was collected from tree devices positioned in different body areas: wrist, chest and ankle. Each device has three sensors embedded: a 3-axis accelerometer, a 3-axis gyroscope and a 3-axis magnetometer.

The PAMAP2 dataset contains 1.926.896 samples of raw sensor data from 9 different users and 18 different activities. The activities executed by the users are divided in basic activities (walking, running, Nordic walking and cycling), posture activities (lying, sitting and standing), everyday activities (ascending and descending stairs), household (ironing and vacuum cleaning) and fitness activities (rope jumping). Also, the users were encouraged to perform optional activities (watching TV, computer work, car driving, folding laundry, house cleaning and playing soccer).

4.2 Experimental Setup

Using the PAMAP2 dataset [15], each ensemble was trained with data from 8 users and tested with one isolated user, not presented in the training process. This approach is called leave-one-user-out.

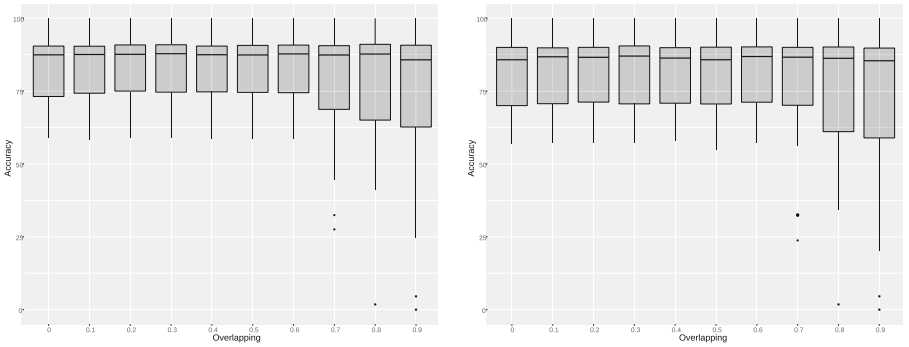
We conducted four experiments with two ensemble models, each one consisting of three classifiers: kNN, Naive Bayes, and Hoeffding Tree (VFDT), as in [8]. As verified in [2], these three classifiers have good classification performance in HAR problems. Thus, we analyze the accuracy performance of one ensemble model with a semi-supervised approach and another ensemble model with a supervised approach.

The box-plots correspond to the variance in accuracy for different values of window size (from 100 to 1000 with increments of 100), overlap factor (from 0.0 to 0.9 with increments of 0.1) and users (from 1 to 9 with increments of 1 user per experiment).

4.3 HAR Accuracy Results

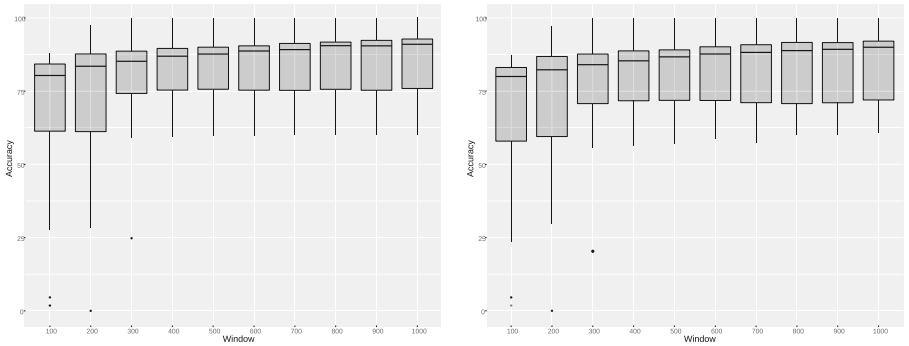
Figure 2 presents the *accuracy* (axis y) for each value of the overlap factor, *overlapping* (axis x). The semi-supervised model reduces accuracy variance, compared with supervised model, for most of the overlapping and has average accuracy close to 90%. For both models, overlapping has more influence on accuracy

[2] http://archive.ics.uci.edu/ml/datasets/pamap2+physical+activity+monitoring.



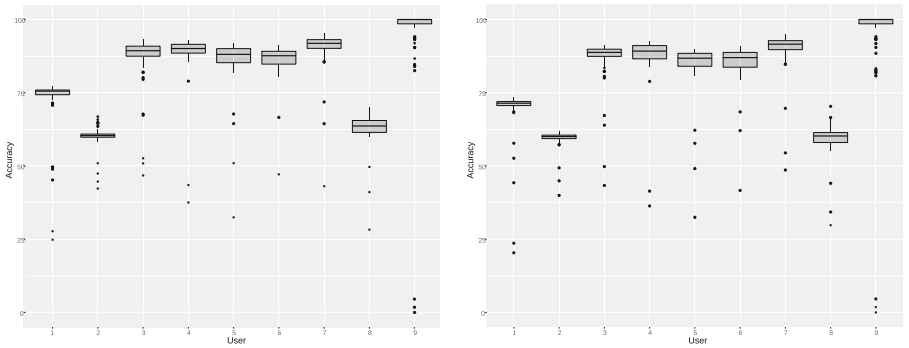
(a) Semi-supervised (b) Supervised

Fig. 2. Supervised vs semi-supervised ensemble accuracy: overlap factor influence.



(a) Semi-supervised (b) supervised

Fig. 3. Supervised vs semi-supervised ensemble accuracy: window size influence.



(a) Semi-supervised (b) Supervised

Fig. 4. Supervised vs semi-supervised ensemble accuracy: user influence.

for values higher than 0.7, but the semi-supervised model is less susceptible to that influence than the supervised model. As shown in Fig. 3, variance of *accuracy* (axis y) and *window size* (axis x), the semi-supervised model reduces accuracy variance for each value of window size. We also notice that windows with small sizes have worse results, especially for sizes of 100 and 200.

We analyze the models accuracy for each user. For that, we analyze the variance of *accuracy* (axis y) when varying the hyperparameters *window size* and *overlapping* for each *user* (axis x). In Fig. 4, for both models, user 5 and 6 have variance higher than users 2 and 1. The semi-supervised model reduces accuracy variance for users 4 and 8. An interesting case to analyze is user 9. For most of the cases, the accuracy is 100%, however user 9 only has instances for the Rope Jumping activity, which means that this user influences the results to higher values. In some cases, the individual accuracy can be lower, as we can see with users 2, 8 and 1, justifying the analyzes by user instead of analyzing all population.

With the results, we can see that window size and overlapping do influence the accuracy of the models. Based on these results and depending on the HAR application, one can decide about the window size and overlap factor level that make possible a certain minimum desired classification accuracy. The exhaustive exploration allows us to also understand the acceptable ranges to explore within a runtime autotuning system, e.g., to keep a minimum accuracy (e.g. 80%). These ranges can be used, at runtime, to search for the best combination of the hyperparameters that provide the best results, e.g., in terms of execution time or energy consumption. The following subsection shows the impact on execution time and energy consumption of different window sizes and overlap factor.

4.4 Execution Time and Energy Consumption

We also analyze the execution time and energy consumption for processing all the data from the PAMAP2 dataset. For that, we conducted experiments in an ODROID-XU+E[3] system running Android. The experiments focus on a single user, user 6, and the execution time and energy required to process 250.096 raw samples.

The first experiment is about the execution time necessary to process all the data of user 6. The execution time was divided into three parts. The first part, *samplingTime*, represents the time required to access all the data from the user and the instantiation of each data window as an instance. The second part, *featureTime* is the feature extraction and the "final instance" instantiation, this part depends on the *window size* and the *overlap* factor used. The last part, *classificationTime* is the total time required to classify all the instances calculated in the feature extraction phase.

In Fig. 5, since the feature extraction depends on the window size, the time to calculate all instances increases as the window size also increases, despite the decreasing number of calculated features. This means that the feature extraction

[3] https://www.hardkernel.com/.

phase is sensitive to the number of raw instances to process. Furthermore, as we increase the overlap factor, due to the increased number of instances that are calculated, the execution time also increases. The classification time is rather small and slightly increases as the number of calculated features augments.

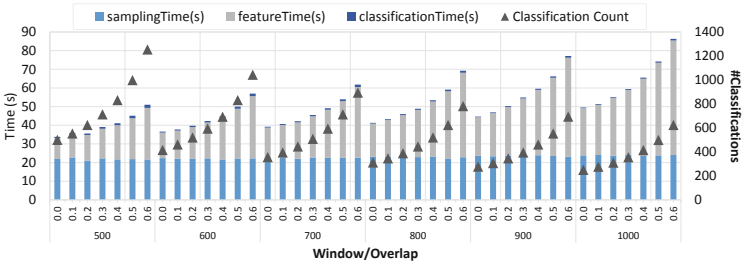


Fig. 5. Total execution time required (left axis) to process the PAMAP2 dataset, per window size and overlap factor, divided in three parts: sampling (data extraction), features extraction and classification. The number of classifications per configuration (right axis) is shown as triangle marks.

The second experiment is presented as a heat map representing the energy, *Joules*, consumed to process raw data from user 6. For the different *window sizes* and *overlap factors*. The colors represent a range of *Joules*, where the red color depicts higher energy consumed and, reversely, green color depicts less energy consumed. The *accuracy* is also shown in the map over each circle depicting the energy color to compare the energy consumed with the classification accuracy.

In Fig. 6, we can see that smaller windows result in less energy consumption than bigger windows. This is due to the increased effort to calculate features for larger window sizes. It is also perceivable that increasing the overlap factor also increases the energy consumed, essentially due to the increased number of feature calculations and classifications to be carried out.

Relating the energy consumption with the accuracy achieved for a given configuration, it is observable that the best accuracy values reside in more "heated" zones, i.e., where energy consumption is higher. Lower window sizes present lower accuracy while higher window sizes provide higher accuracy. For instance, in configurations without overlapping (i.e., with an overlap factor of 0), the accuracy rises from 85% for a window of size 500 to 90% for a window of size 1000.

The overlap shows more fluctuations in terms of accuracy, however with the best factors concentrated between 0.1 and 0.5. This shows that it is not trivial to select a single window size and overlap factor if it is intended to have two possible scenarios, one where accuracy is the most important factor and another one where energy consumption is the top priority but still with a minimum accuracy value in mind.

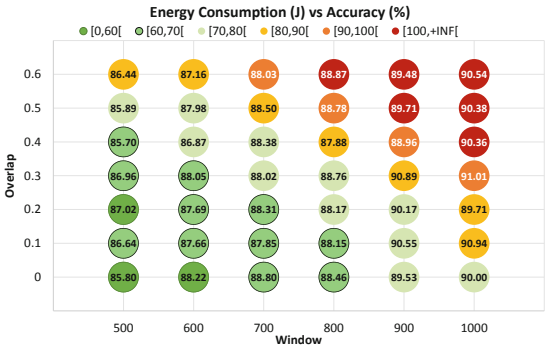


Fig. 6. Energy consumed, in Joules, while processing the PAMAP2 dataset, per window size and overlap factor. Green values represent less energy consumed while red values represent higher energy consumed. The values in each configuration represent the accuracy, in percentage, of that configuration.

5 Conclusion

In this work we presented an analysis of the impact of hyperparameters, as window size and overlap factor, on HAR classification accuracy, execution time and energy consumption. The analysis was focused on a public dataset, which includes raw sensor data from 9 different users and 18 physical activities.

The experimental results confirm the need of adapting the classification model to the current user. Due to the impact of window size and overlap factor, each activity requires a specific configuration of these hyperparameters in order to improve classification accuracy.

Furthermore, the results also motivate the development of a system that is able to adapt the application at runtime when trade-offs between performance accuracy and energy consumption need to be considered. Bearing in mind this, the window size and overlap factor can be used to develop runtime strategies able to adapt these parameters according to the target goals.

As future work, we plan to implement a system able to dynamically adjust at runtime the window size and overlap factor and aware of activities and users. The dynamic adaptation needs to consider an exploration of possible parameter configurations to find the best configurations for each adaptation scenario and thus the experimental results presented in this paper are also part of that exploration phase.

References

1. Krishnan, N.C., Cook, D.J.: Activity recognition on streaming sensor data. Pervasive Mob. Comput. **10**, 138–154 (2014)
2. Dobbins, C., Rawassizadeh, R., Momeni, E.: Detecting physical activity within lifelogs towards preventing obesity and aiding ambient assisted living. Neurocomputing **230**, 110–132 (2017)

3. Miluzzo, E., Varshavsky, A., Balakrishnan, S., Choudhury, R.R.: TapPrints: your finger taps have fingerprints. In: MobiSys (2012)
4. Aggarwal, J.K., Ryoo, M.S.: Human activity analysis: a review. ACM Comput. Surv. **43**(3), 1–43 (2011). Article ID 16
5. Lara, O.D., Labrador, M.A.: A survey on human activity recognition using wearable sensors. IEEE Commun. Surv. Tutor. **15**(3), 1192–1209 (2013)
6. Ramamurthy, S.R., Roy, N.: Recent trends in machine learning for human activity recognition - a survey. Wiley Interdisc. Rev. Data Min. Knowl. Discov. **8**(4), e1254 (2018)
7. Shoaib, M., Bosch, S., Incel, O., Scholten, H., Havinga, P.: A survey of online activity recognition using mobile phones. Sensors **15**, 2059–2085 (2015)
8. Cardoso, H., Mendes-Moreira, J.: Improving human activity classification through online semi-supervised learning. In: Workshop StreamEvolv Co-located with ECML/PKDD 2016, pp. 15–26 (2016)
9. Pisani, P.H., Lorena, A.C.: A systematic review on keystroke dynamics. J. Braz. Comput. Soc. **19**(4), 573–587 (2013)
10. Mannini, A., et al.: Activity recognition using a single accelerometer placed at the wrist or ankle. Med. Sci. Sports Exerc. **45**(11), 2193 (2013)
11. Banos, O., et al.: Window size impact in human activity recognition. Sensors **14**(4), 6474–6499 (2014)
12. Harasimowicz, A., Dziubich, T., Brzeski, A.: Accelerometer-based human activity recognition and the impact of the sample size. In: Proceedings of the 13th International Conference on Artificial Intelligence, Knowledge Engineering and Data Bases, Gdansk, Poland (2014)
13. Baños, O., et al.: Evaluating the effects of signal segmentation on activity recognition. In: IWBBIO (2014)
14. Niazi, A.H., et al.: Statistical analysis of window sizes and sampling rates in human activity recognition. In: HEALTHINF (2017)
15. Reiss, A., Stricker, D.: Creating and benchmarking a new dataset for physical activity monitoring. In: Proceedings of the 5th International Conference on PErvasive Technologies Related to Assistive Environments. ACM (2012)
16. Figo, D., Diniz, P.C., Ferreira, D.R., Cardoso, J.M.P.: Preprocessing techniques for context recognition from accelerometer data. Pers. Ubiquitous Comput. **14**(7), 645–662 (2010)
17. Zhou, Y., Goldman, S.: Democratic co-learning. In: 16th IEEE International Conference on Tools with Artificial Intelligence, pp. 594–602. IEEE Computer Society (2004)
18. Zhou, Z.H., Li, M.: Tri-training: exploiting unlabeled data using three classifiers. IEEE Trans. Knowl. Data Eng. **17**(11), 1529–1541 (2005)