Human Activity Recognition using Time Series Feature Extraction and Active Learning

Vangjel, VK, Kazllarof
Department of Mathematics, University of Patras
vkazlarof@upatras.gr

Sotiris, SK, Kotsiantis
Department of Mathematics, University of Patras
kotsiantis@upatras.gr

ABSTRACT

Today, portable devices like smartwatches and smartphones have made a great impact in human's wellbeing. From sleep monitoring to exercise scheduling, Human Activity Recognition had played a major role in the habits of the people. In this work, we exploit a Time Series dataset that describes a Human Activity Recognition signal. In the beginning, we extract the features oriented on Spectral, Statistical and Temporal domains. Then, we construct a dataset for each domain and we calculate the classification results using a number of different classifiers. In the sequel, we apply Active Learning techniques and calculate their classification accuracy performance using a small portion of the original datasets as initial labeled set. Finally, we compare the original results with the ones produced with Active Learning methods.

CCS CONCEPTS

• **Computing methodologies** → Machine learning; Learning settings; Active learning settings; • **Mathematics of computing** → Probability and statistics; Statistical paradigms; Time series analysis; • **Human-centered computing** → Ubiquitous and mobile computing; Ubiquitous and mobile devices.

KEYWORDS

Machine Learning, Active Learning methods, Activity Recognition, Feature Extraction

ACM Reference Format:
Vangjel, VK, Kazllarof and Sotiris, SK, Kotsiantis. 2022. Human Activity Recognition using Time Series Feature Extraction and Active Learning. In *12th Hellenic Conference on Artificial Intelligence (SETN 2022), September 07–09, 2022, Corfu, Greece.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3549737.3549787

1 INTRODUCTION

Accelerometer, gyroscope and other inertial sensors on portable or wearable devices have produced a great amount of data for Human Activity Recognition (HAR) [3]. Collecting and then classifying this data have been a major study field of Time Series Analysis [16].

Feature extraction methods have been widely used for the classification of the data. With these methods, new datasets are created based the source Time Series and they consists of numerical or nominal features that describe a specific domain in the Time Series space. In the sequel, a feature pruning is conducted in order to keep the most relevant one [1].

On one hand, we have plenty of data collected from all these devices. On the other hand, finding the label of the exact activity that the Time Series sequence is describing is something that in most cases it is very difficult and expensive, due to the limitation that experimenters need to be actively involved and label each signal produced by the sensors. Active Learning (AL) methods can solve this problem by using a small amount of initial labeled data and create a model using this data. Then, this model is used to rank all the newly produced signals and the most informative ones are chosen and provided to the experimenter in order to label them. This way, the amount of work and cost is highly reduced while keeping a satisfying classification performance [14].

In the following sections, we describe the dataset creation process, we demonstrate the experiments conducted in the datasets and then we analyze the results.

2 RELATED WORK

HAR have been a well-studied field in the recent years due to the evolution of the smart wearable and portable devices. In [2] a HAR framework has been proposed that is based on feature extraction and feature selection techniques using 3-dimensional accelerometer sensors and extracting features in time, statistical and frequency domains, showing some promising results. As AL is adopted in the research, many studies have been conducted for HAR under AL scheme [10]. In [11] a semi-supervised classifier combined with a novel Bayesian stream-based active learning is proposed using a three-module architecture composed of a feature extractor. Dynamic AL has been used in [2] in which a method that not only selects informative samples but also identifies new activities that are not included in the initial labeled set. Moreover, a study using smartwatch devices for HAR with the use of AL can be found in [15] achieving 93.3% accuracy across 12 participants. Another approach has been conducted in [17] where a self-trained model has been proposed, in which the unlabeled data is used in the learning process along with the labeled one without the involvement of the experimenter.

3 DATA DESCRIPTION

The data is from Human Activity Recognition Using Smartphones Data Set published in UCI repository [5]. In the experiments, a group of 30 volunteers has been recorded on activities of walking, walking upstairs, walking downstairs, sitting, standing and laying wearing a smartphone on their waist. The dataset produced consists of inertial signals by the accelerometer and the gyroscope in the

SETN 2022, September 07–09, 2022, Corfu, Greece

ACM ISBN 978-1-4503-9597-7/22/09…$15.00
https://doi.org/10.1145/3549737.3549787

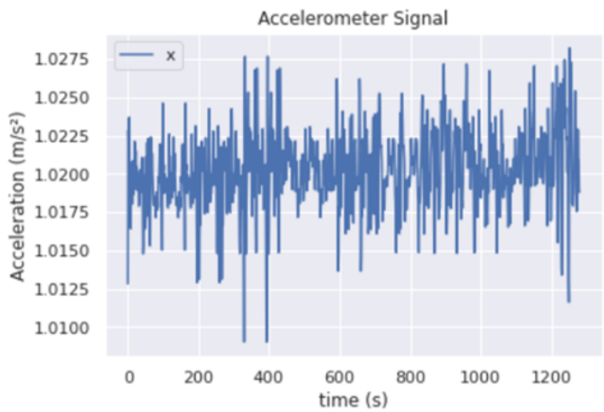

Figure 1: Signal graph of total acceleration produced by the accelerometer on the x-axis

3-dimensional space over time. The measurements provide the total triaxial acceleration of the volunteer and the estimated body acceleration from the accelerometer along with the triaxial angular velocity for the body from the gyroscope. The signals have been then processed for noise removal and sampled in fixed width sliding windows of 2.56 seconds and 50% overlap resulting to 128 readings per window. From each window, a vector of features have been produced on time and frequency domains [7].

For our experiments, we chose the signal produced by the volunteer's total acceleration on the x-axis. For train dataset, 70% the original dataset is used and it consists of 7352 instances and 128 features. The rest 30% is used for testing and it consists of 2947 instances [7]. In Figure 1 is presented the graph of the signal over time.

From the signal, a number of features are extracted on three different domains. The domains are oriented around Spectral features that are used for discovering underlying periodicities, Statistical features that are used for the statistical analysis of the signal and Temporal features that are used for analyzing temporal windows of the signal. Every domain obtains a dataset with same number of instances but different number of features. Spectral features obtain a dataset of 145 attributes, while Statistical and Temporal features obtain datasets of 36 and 18 features respectively. Some of the selected features are FFT mean coefficient, Fundamental frequency and Human range energy from Spectral domain. From the Statistical domain, there are features like Histogram, Kurtosis and Skewness while from the Temporal domain, Absolute energy, Centroid and Entropy can be found. Due to lack of space, the complete list of the extracted features can be found in [1].

Next, the feature normalization follows. In this step, the highly correlated and the low variant features of each dataset are removed. In addition, a standardization of the datasets is applied by centering the data on the mean and then scaling it to unit variance. The process above produces a final dataset of 102, 16 and 11 features for Spectral, Statistical and Temporal domains respectively.

The feature extraction and normalization is conducted using Time Series Feature Extraction Library (TSFEL) written in Python language. It is an experimental package for extracting features in Spectral, Statistical and Temporal domains [1].

4 EXPERIMENTS

In this work, a number of classifiers like Gaussian Naïve Bayes, KNN, Decision Trees, Random Forest, Bagging, Gradient Boosting (GDB) and Multilayer Perceptron (MLP) is studied, with the latest two being the most performant, thus they will be presented in the current work. GDB is a generalization of boosting technique by allowing optimization of an arbitrary differentiable loss function using decision trees as weak learners [6]. MLP is feedforward neural network that utilizes an activation function on each neuron, except input nodes and optimizes its coefficients using backpropagation technique [9].

In Table 1 we demonstrate the classification accuracy of the classifiers described above for the datasets obtained by each feature domain and they will be used as benchmarks for comparing the results of the AL process.

From the results, it is observed that datasets created with features from Spectral domain show the best classification accuracy for both GDB and MLP. Moreover, GDB outperforms MLP on Spectral and Statistical domains but underperforms for a small amount on the Temporal domain.

For the AL process, there are several techniques for querying the unlabeled instances. In this work, the Pool Based technique is used in which each dataset is separated a small Labeled pool (L) where each instance has its classification label and into a much bigger Unlabeled pool (U) where instances have no classification label. Using the L to create an initial model, the instances in U are ranked based on the information they provide to the model using a ranking method. After the ranking process, a batch of the highest ranked instances is provided to the experimenter to label them. Afterwards, the in newly labeled instances are added to the L pool and the model is retrained, ending an AL cycle [13].

For the ranking process, we use Uncertainty Sampling techniques in which the instances with the highest rank are the ones that the model is most uncertain. For measuring the uncertainty we use Entropy, Smallest Margin and Least Confidence methods, with the latter showing the best results. Its formula is described in the following equation:

$$x_{LC} = \arg \max_{x} \ 1 - P_{\theta}(y|x) \tag{1}$$

where LC refers to the instance x the with the Least Confident probability on the U pool given that it is classified to the y class by the model [13].

In the sequel, we define the Labeled Ratio (R) as the initial percentage of the L compared to the whole dataset, and it is describe by the following equation:

$$R = \frac{L}{L+U} * 100 \tag{2}$$

For our experiments, we studied the behavior in for initial R% of 5%, 10%, 15% and 20% of the dataset. In each AL cycle, a dynamic sized batch of most informative instances is selected. The size is calculated in a way that the L pool will double in size in the last

Table 1: Classification results for GDB and MLP for datasets obtained by Spectral, Statistical and Temporal domains

Domain	Gradient Boosting	Multilayer Perceptron
Spectral	81.98%	80.01%
Statistical	75.84%	75.06%
Temporal	75.09%	75.23%

Table 2: Classification results of GDB and MLP classifiers with LC and RS methods for datasets obtained by Spectral, Statistical and Temporal domains

Domain	GDB-LC	GDB-RS	MLP-LC	MLP-RS
Spectral	81.71%	81.64%	77.77%	78.96%
Statistical	75.64%	73.70%	76.45%	76.11%
Temporal	75.45%	75.84%	74.58%	74.55%

iteration being 10%, 20%, 30% and 40% of the dataset. The AL process stops after conducting 15 iterations of AL cycles. The model produced in the last iteration, is used evaluated then in the test set.

For the experiments, we used the classifiers of Scikit Learn [12] library along with Libact [18] library for the AL methods, both written in Python Language.

5 RESULTS

Although the experiments are conducted in an initial R% of 5% and greater, there isn't great improvement in the classification accuracy that justifies the bigger initial labeled pool. Moreover, focusing on smaller initial L pools, we approach scenarios that are more realistic to real life.

On Table 2 we demonstrate the classification results of the last constructed model for GDB and MLP for the feature domains mentioned above using Least Confidence (LC) method in comparison with Random Sampling (RS) that doesn't do anything sophisticating than selecting instances from the U pool randomly.

From the results, we notice that we can achieve quite similar results with only a small amount of initial labeled instances. Another noticeable observation is that although generally LC shows better results than RS, in some cases RS has shown similar results or outperform LC showing that the obtained datasets come with a lot of noise meaning that just choosing a subset of it, we can achieve better classification results. Lastly, we observe that in some cases the AL process outperforms the benchmark results. More specifically, the dataset obtained by the Statistical features domain is outperformed by MLP on the AL scheme and the one obtained by the Temporal feature domain is outperformed by GDB.

In Figure 2 is demonstrated the learning curve for Spectral feature domain while the L pool grows. In Figure 3 and Figure 4 are demonstrated the learning curve for Statistical and Temporal domains respectively.

From the results, it is observed that the clearest upwards trend happens with GDB using LC method for Spectral feature domain and MLP with LC for Statistical feature domain. For Temporal domain, GDB shows upwards trends in both LC and Random sampling but with LC spiking in the last iterations as the number of instances increase, meaning that adding more instances will most probably outperform the RS method.

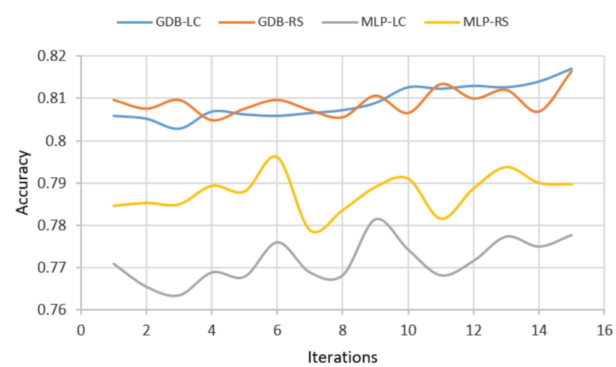

Figure 2: Learning curve of GDB and MLP using LC and RS methods for the Spectral feature domain

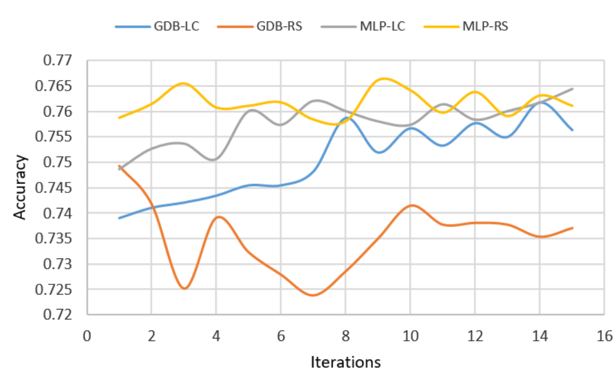

Figure 3: Learning curve of GDB and MLP using LC and RS methods for Statistical feature domain

Compared to similar studies like [10], our approach shows better performance in classification accuracy using a smaller initial labeled set with the use of state-of-the-art classifiers.

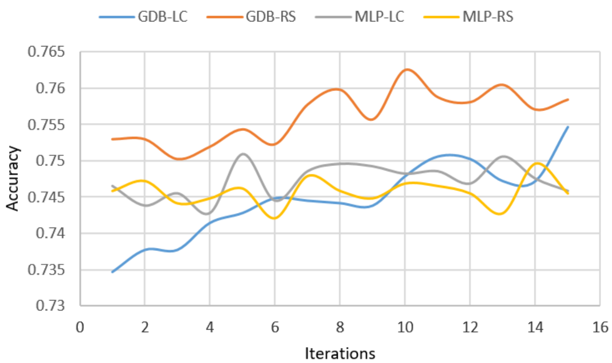

Figure 4: Learning curve of GDB and MLP using LC and RS methods for Temporal feature domain

6 CONCLUSION AND FUTURE WORK

In this work, we conducted AL experiments on a HAR dataset that describes the total acceleration on the x-axis by extracting features on Spectral, Statistical and Temporal domains. We used GDB and MLP as classifiers with LC method along with RS query strategies in order to obtain the most informative instances with an initial labeled size of R=5% and we compared the results with the benchmark results that that all instances are labeled.

From the results, it is observed that similar classification accuracy can be achieved with a much smaller initial labeled dataset. In addition, the superiority of active learning methods is shown compared to random sampling, although in some cases random sampling shows better performance demonstrating the amount of noise that the produced datasets can have.

In the future, we plan to use hyper parameters for GDB and MLP in order to tune them in order to perform better and have better benchmark and AL results. In addition, we would also like to test more sophisticated algorithms like Convolutional Neural Networks that had shown promising results in the literature in the HAR problem [4] under AL scheme. Moreover, another approach we would like to add is using a noisy oracle, meaning that the experimenter will have an error rate in finding the correct label [8]. Finally, we will study a combination of self-train and active learning in which the experimenter will have an even smaller participation in the learning process, because the model will label a great amount of unlabeled instances using self-train.

REFERENCES

[1] Marília Barandas, Duarte Folgado, Letícia Fernandes, Sara Santos, Mariana Abreu, Patrícia Bota, Hui Liu, Tanja Schultz, and Hugo Gamboa. 2020. TSFEL: Time Series Feature Extraction Library. SoftwareX 11, (January 2020), 100456. DOI:https://doi.org/10.1016/J.SOFTX.2020.100456

[2] Haixia Bi, Miquel Perello-Nieto, Raul Santos-Rodriguez, and Peter Flach. 2021. Human Activity Recognition Based on Dynamic Active Learning. IEEE J. Biomed. Heal. Informatics 25, 4 (April 2021), 922–934. DOI:https://doi.org/10.1109/JBHI.2020.3013403

[3] Andreas Bulling, Ulf Blanke, and Bernt Schiele. 2014. A tutorial on human activity recognition using body-worn inertial sensors. ACM Comput. Surv. 46, 3 (January 2014). DOI:https://doi.org/10.1145/2499621

[4] Federico Cruciani, Anastasios Vafeiadis, Chris Nugent, Ian Cleland, Paul McCullagh, Konstantinos Votis, Dimitrios Giakoumis, Dimitrios Tzovaras, Liming Chen, and Raouf Hamzaoui. 2020. Feature learning for Human Activity Recognition using Convolutional Neural Networks: A case study for Inertial Measurement Unit and audio data. CCF Trans. Pervasive Comput. Interact. 2, 1 (March 2020), 18–32. DOI:https://doi.org/10.1007/S42486-020-00026-2/TABLES/6

[5] Dheeru Dua and Casey Graff. 2019. UCI Machine Learning Repository. University of California, Irvine, School of Information and Computer Sciences. Retrieved March 21, 2021 from http://archive.ics.uci.edu/ml

[6] Jerome H. Friedman. 2001. Greedy function approximation: A gradient boosting machine. Ann. Stat. 29, 5 (October 2001), 1189–1232. DOI:https://doi.org/10.1214/AOS/1013203451

[7] Daniel Garcia-Gonzalez, Daniel Rivero, Enrique Fernandez-Blanco, and Miguel R. Luaces. 2020. A Public Domain Dataset for Real-Life Human Activity Recognition Using Smartphone Sensors. Sensors 20, 8 (April 2020), 2200. DOI:https://doi.org/10.3390/S20082200

[8] Gaurav Gupta, Anit Kumar Sahu, and Wan Yi Lin. 2019. Noisy Batch Active Learning with Deterministic Annealing. (September 2019). Retrieved March 13, 2021 from http://arxiv.org/abs/1909.12473

[9] SIMON HAYKIN. 2000. Neural Networks: A Guided Tour. Soft Comput. Intell. Syst. (January 2000), 71–80. DOI:https://doi.org/10.1016/B978-012646490-0/50007-X

[10] Rong Liu, Ting Chen, and Lu Huang. 2010. Research on human activity recognition based on active learning. 2010 Int. Conf. Mach. Learn. Cybern. ICMLC 2010 1, (2010), 285–290. DOI:https://doi.org/10.1109/ICMLC.2010.5581050

[11] Saad Mohamad, Moamar Sayed-Mouchaweh, and Abdelhamid Bouchachia. 2020. Online active learning for human activity recognition from sensory data streams. Neurocomputing 390, (May 2020), 341–358. DOI:https://doi.org/10.1016/J.NEUCOM.2019.08.092

[12] Fabian Pedregosa, Gael Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. 2012. Scikit-learn: Machine Learning in Python. J. Mach. Learn. Res. 12, (January 2012), 2825–2830. DOI:https://doi.org/10.48550/arxiv.1201.0490

[13] Burr Settles. 2012. Active learning. Synth. Lect. Artif. Intell. Mach. Learn. 18, (June 2012), 1–111. DOI:https://doi.org/10.2200/S00429ED1V01Y201207AIM018

[14] Burr Settles, Mark Craven, and Lewis Friedl. 2008. Active Learning with Real Annotation Costs. In Proceedings of the NIPS Workshop on Cost-Sensitive Learning, 1–10.

[15] Farhad Shahmohammadi, Anahita Hosseini, Christine E. King, and Majid Sarrafzadeh. 2017. Smartwatch Based Activity Recognition Using Active Learning. Proc. - 2017 IEEE 2nd Int. Conf. Connect. Heal. Appl. Syst. Eng. Technol. CHASE 2017 (August 2017), 321–329. DOI:https://doi.org/10.1109/CHASE.2017.115

[16] Marcin Straczkiewicz, Peter James, and Jukka Pekka Onnela. 2021. A systematic review of smartphone-based human activity recognition methods for health research. npj Digit. Med. 4, 1 (October 2021), 1–15. DOI:https://doi.org/10.1038/s41746-021-00514-4

[17] Chi Ian Tang, Ignacio Perez-Pozuelo, Dimitris Spathis, Soren Brage, Nick Wareham, and Cecilia Mascolo. 2021. SelfHAR: Improving Human Activity Recognition through Self-training with Unlabeled Data. Proc. ACM Interactive, Mobile, Wearable Ubiquitous Technol. 5, 1 (March 2021), 30. DOI:https://doi.org/10.1145/3448112

[18] Yao-Yuan Yang, Shao-Chuan Lee, Yu-An Chung, Tung-En Wu, Si-An Chen, and Hsuan-Tien Lin. 2017. libact: Pool-based Active Learning in Python. DOI:https://doi.org/10.48550/arxiv.1710.00379