[TO BE DEFINED]

Matteo Drago, Riccardo Lincetto[†]

***Abstract*—With the increasing interest in deep learning techniques and its applications, also Human Activity Recognition (HAR) saw significant improvements; before neural networks were put into practice, most of the research activities on the field relied on hand-crafted features which, however, couldn't represent nor distinguish well enough complex and articulated movements. Moreover, the use of smart devices and wearable sensors brought the challenge to another level: dealing with high-dimensional and noisy time series while assuring optimal performances requires a detailed study and, most of all, a considerable computational effort.**

In our paper we present the design of an HAR architecture which implements convolutional layers in order to extract significant features from windows of samples, along with Long-Short Term Memory (LSTM) layers, suitable to exploit time dependencies among consecutive samples. For our study, we designed the system in order to minimize the collection of layers per network and thus the amount of parameters to train, which could be of great advantage in real time application. In addition, we also decided to study how performances change if we split the process into two distinct phases: the first one that performs *activity detection* while the last one *activity classification*. The dataset that we used to assess the efficiency of our architecture is the OPPORTUNITY dataset.

***Index Terms*—Human Activity Recognition, Machine Learning, Neural Networks, Motion Detection.**

I. Introduction

During the past decade, time series classification has captured growing interest thanks to the introduction of deep learning mechanisms, such as neural networks. These tools are indeed capable of identifying and learning signal features, which are then exploited for classification, without the need of human domain-knowledge: this is a huge step forward considering that features were traditionally hand-crafted. Human Activity Recognition (HAR) in particular has been fostered by the spread of powerful, efficient and affordable sensors, which nowadays are commonly found in mobile phones and wearable devices, with multiple applications, ranging from health care to gaming and virtual reality [1]. Wearable sensors allow us to collect and process a huge amount of signals, which are essential for deep neural networks (DNN) to work properly: in fact, in order for them to learn and for being accurate enough to be preferred over standard machine learning approaches, we need the input training set to be heterogeneous, meaningful and representative of the problem. For these reasons, HAR is not an easy classification problem: when dealing with on-body sensors, system performances heavily depends on human behaviour, which is a source of high variability; moreover, data collected from sensors is typically high-dimensional, multimodal and subjected to noise, making the problem even more difficult from a machine learning perspective.

In the recent years, several ways of performing activity detection and classification have been proposed: in the literature there's no shortage of models. The trend has been to expand the power of networks, adding more and more layers: this resulted in more accurate models, that had to face though an increasing computational complexity. However, specifically when dealing with real time applications, computational power is limited and the possibility of using too complex models is far from being realizable. Moreover, as pointed out in [2] and [3], despite the proliferation of models to perform activity detection and classification, the lack of common data to perform a baseline evaluation and of structured and fixed implementation details prevented a fair comparison between different solutions.

Considering than that the activity recognition problem has been already widely addressed by many authors, we decided to present a systematic comparison between two different commonly proposed types of pipeline. The two differ on how inactivity is handled: the first one tries to learn a representation of the signals where no action is performed, adding a null class to the other activities; the second one instead splits the classification into two tasks, first deploying an activity detector that filters out inactivity signals and then classifying the remaining activities. Our study is meant to provide a baseline for future work, giving an idea on which system could be more appealing. In order to assess the efficiency of our models, that have been designed against the trend trying to minimize the number of trainable parameters, we used the **OPPORTUNITY** dataset [2], [4] which will be described in details in the following sections.

In conclusion, the contributions of this paper are:

- overview of the latest progresses of the state of the art;
- implementation of those solutions;
- comparison of two different approaches.

The paper is organized as follow: section II provides a summary of the latest and more important works related to our studies; in section III we start delving into the details of how we organized our HAR architecture, step by step; section IV is dedicated to the description of the dataset and to the decisions we made in the preprocessing phase; finally in section V we are ready to describe meticulously the learning framework, while sections VI and VII are for discussion of results and for drawing our conclusions.

[†] Department of Information Engineering, email: {matteo.drago,riccardo.lincetto}@studenti.unipd.it

II. Related Work

The **OPPORTUNITY** activity recognition dataset has been introduced in [4] to overcome the lack of an evaluation setup, to compare different classification systems and to provide a more exhaustive dataset compared to the others, which "are not sufficiently rich to investigate opportunistic activity recognition, where a high number of sensors is required on the body, in objects and in the environment, with a high number of activity instances". As pointed out in [2] in fact, previously, several datasets were related to the activities which were to be classified: this is due to researchers acquiring signals only from sensors located in specific locations, according to the task of interest. To overcome this drawback, the **OPPORTUNITY** dataset has been gathered from a monitored, sensor rich environment aggiungere img dell'ambiente? : objects from the scene were connected to acquisition sensors, while people participating to the session were equipped with on-body sensors; signals collected from different sensors will be described in section IV. This particular dataset has been fundamental over the past years, it provided indeed an heterogeneous and complete set of time series, perfectly suitable for different studies in the **HAR** domain. In [2] they present it as a *benchmark dataset*; as a demonstration, they provide the results obtained with four classification techniques (*k-nearest neighbours, nearest centroid, linear discriminant analysis, quadratic discriminant analysis*) and they compare them with other works that used the same dataset. inseriamo anche i valori che ottengono nel paper per confronto?

The authors in [5] proposed an exhaustive framework which, besides the standard preprocessing on the activity data sequence (filling of the gaps via interpolation and data normalization), presents also a solution for the well-known class imbalance problem [6]. Moreover, they also include a post-processing procedure after classification consisting of a smoothing operation along the temporal axis (i dati non vengono finestrati e quindi loro li filtrano) and of a strategic fusion procedure to integrate prediction sequences from different classifiers, in order to reduce the risk of making an erroneous classification. The classifiers used in this work consisted in a 1-layer neural network (1NN) and a Support Vector Machine (SVM, complete overview of this tool in [7]). Even for this work the **OPPORTUNITY** dataset has been used for assessing performances.

As we outlined before, the recent explosion of

III. Processing Pipeline

I would start the technical description with a *high level* introduction of your processing pipeline. Here you do not have to necessarily go into the technical details of every processing block, this will be done later as the paper develops. What I would like to see here is a description of the general approach, i.e., which processing blocks you used, how these were concatenated, etc. A diagram usually helps.

We start off our analysis by preprocessing the collected signals within the MATLAB framework: we chose that because it makes it simple to deal with matrices. What we do in this first step is then to import the data collected by sensors, which are given as .dat files, select the signals from on-body sensors and discard the others, replacing the missing values by means of interpolation and, at last, store them as .mat files. What we do next is to import the stored data, this time using python, and prepare the matrices for the classification task: this consists of concatenating the data, segmenting it into windows, scaling the signals and other common steps. Once the data is ready to be classified, a model is defined and trained on the available data. This is done for both the locomotion activity and gestures recognition, i.e. with two different sets of labels. This system, which is forced to learn also the null class together with the actual movements, is then compared to a different system where two models are deployed: the first one has the only purpose of detecting activity, while the second classifies the activity, if present.

IV. Signals and Features

Being a machine learning paper, I would put here a section describing the signals you have been working on. If possible, you should describe, in order, 1) the measurement setup, 2) how the signals were pre-processed (to remove noise, artifacts, fill gaps or represent them through a constant sampling rate, etc.). After this, you should describe how *feature vectors* were obtained from the pre-processed signals. If signals are *time series* this also implies stating the segmentation / windowing strategy that was adopted, to then describe how you obtained a feature vector for each time window. Also, if you also experiment with previous feature extraction approaches, you may want to list them as well, in addition to (and before) your own (possibly new) proposal.

The signals that we use to perform HAR are the ones collected in the OPPORTUNITY activity recognition dataset. The measurement setup is then the one presented in [**?**] and [**?**]. Our analysis though is based only on on-body sensor signals, which means that we kept the signals of only a subset of the available sensors: discarding the other signals then, we ended up with 113 signals. During the preprocessing, we discarded also 3 of them, belonging to the same physical device, because there weren't any measurements in most of the cases. This led us to work on 110 signals. Since we noticed that the almost all the sensors, at the beginning and at the end of the measurement sessions, have sequences where there isn't any sample recorded, we decided to discard the head and the tail of each session, in such a way that we start and stop with all the measurements being registered. This choice was made also to facilitate interpolation. In MATLAB we perform a splines interpolation, which uses a cubic polynomial. The decision of cutting head and tail prevented our code from interpolating a piece of signal which has only one "edge". Then, to perform classification on the data of one subject, we stacked sessions ADL 1 to 3 and Drill to create our training set, and then ADL4 and ADL5 as test set. In some cases, interpolation leaves entire columns to NaN because it isn't provided any data to interpolate those values. We solved the problem by setting to 0 those entire columns. Subsequently we scaled the signals by subtracting their means and dividing by their variance (or sigma?). After this, data is shaped into windows of 15 samples (500 ms) with a stride of 5 samples. The approach that we used to segment the data was then the sliding window introduced above. To perform classification, though we had to assign to each window a unique label, which we decided to be corresponding to the label present with more samples. This doesn't constitute a problem per se, even when changing the size of the sliding window, as long as it is kept short enough and ...

V. Learning Framework

Here you finally describe the learning strategy / algorithm that you conceived and used to solve the problem at stake. A good diagram to exemplify how learning is carried out is often very useful. In this section, you should describe the learning model, its parameters, any optimization over a given parameter set, etc. You can organize this section in sub-sections. You are free to choose the most appropriate structure.

One of the main problems in Human Activity Recognition is handling inactivity.

Thinking of a real recognition system, In this paper we compare two different learning strategies, mimicking a real system. In the first V-A, One Shot Classification, the model is trained to learn a representation of the involved classes together with the null class

A. One Shot Classification

B. Two Steps Classification

VI. Results

VII. Concluding Remarks

References

[1] O. D. Lara and M. A. Labrador, "A survey on human activity recognition using wearable sensors," *IEEE Communications Surveys Tutorials*, vol. 15, pp. 1192–1209, Third 2013.

[2] R. Chavarriaga, H. Sagha, A. Calatroni, S. T. Digumarti, G. Tröster, J. del R. Millán, and D. Roggen, "The opportunity challenge: A benchmark database for on-body sensor-based activity recognition," *Pattern Recognition Letters*, 2013.

[3] F. Li, K. Shirahama, M. A. Nisar, L. Kping, and M. Grzegorzek, "Comparison of feature learning methods for human activity recognition using wearable sensors," *Sensors*, vol. 18, no. 2, 2018.

[4] D. Roggen, A. Calatroni, M. Rossi, T. Holleczek, K. Frster, G. Trster, P. Lukowicz, D. Bannach, G. Pirkl, A. Ferscha, J. Doppler, C. Holzmann, M. Kurz, G. Holl, R. Chavarriaga, H. Sagha, H. Bayati, M. Creatura, and J. d. R. Mill "Collecting complex activity datasets in highly rich networked sensor environments," in *2010 Seventh International Conference on Networked Sensing Systems (INSS)*, pp. 233–240, June 2010.

[5] H. Cao, M. N. Nguyen, C. Phua, S. Krishnaswamy, and X. Li, "An integrated framework for human activity classification.," in *UbiComp*, pp. 331–340, 2012.

[6] N. Japkowicz and S. Stephen, "The class imbalance problem: A systematic study," *Intelligent data analysis*, vol. 6, no. 5, pp. 429–449, 2002.

[7] M. A. Hearst, S. T. Dumais, E. Osuna, J. Platt, and B. Scholkopf, "Support vector machines," *IEEE Intelligent Systems and their applications*, vol. 13, no. 4, pp. 18–28, 1998.