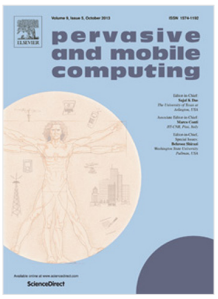
Journal Pre-proof



Lightweight accurate trigger to reduce power consumption in sensor-based continuous human activity recognition

Emanuele Lattanzi, Lorenzo Calisti, Paolo Capellacci

PII: S1574-1192(23)00106-2
DOI: https://doi.org/10.1016/j.pmcj.2023.101848
Reference: PMCJ 101848

To appear in: *Pervasive and Mobile Computing*

Received date : 28 April 2023
Revised date : 29 August 2023
Accepted date : 2 October 2023

Please cite this article as: E. Lattanzi, L. Calisti and P. Capellacci, Lightweight accurate trigger to reduce power consumption in sensor-based continuous human activity recognition, *Pervasive and Mobile Computing* (2023), doi: https://doi.org/10.1016/j.pmcj.2023.101848.

Lightweight Accurate Trigger to Reduce Power Consumption in Sensor-based Continuous Human Activity Recognition

Emanuele Lattanzi[a,*], Lorenzo Calisti[a], Paolo Capellacci[a]

[a]*Department of Pure and Applied Sciences, University of Urbino, Piazza della Repubblica 13, Urbino, 61029, Italy*

Abstract

Wearable devices have become increasingly popular in recent years, and they offer a great opportunity for sensor-based continuous human activity recognition in real-world scenarios. However, one of the major challenges is their limited battery life. In this study, we propose an energy-aware human activity recognition framework for wearable devices based on a lightweight accurate trigger. The trigger acts as a binary classifier capable of recognizing, with maximum accuracy, the presence or absence of one of the interesting activities in the real-time input signal and it is responsible for starting the energy-intensive classification procedure only when needed. The measurement results conducted on a real wearable device show that the proposed approach can reduce energy consumption by up to 95% in realistic case studies, with a cost of performance deterioration of at most 1% or 2% compared to the traditional energy-intensive classification strategy.

Keywords: power-aware machine learning, ubiquitous motion tracking, human activity recognition, wearable devices

1. Introduction

Sensor-based continuous Human Activity Recognition (HAR) has become increasingly popular today thanks to the explosion of the market for wearable devices, which ranges from the most complex smartphones to smartwatches

*Corresponding author
Email address: emanuele.lattanzi@uniurb.it (Emanuele Lattanzi)

up to the simplest pedometers. These types of devices, which accompany us in everyday life, are equipped with different sensors such as accelerometers, gyroscopes, magnetometers, heart rate monitors, etc. which allow us to constantly recognize what kind of activity we are doing. To use the parallelism with "the Era of Ubiquitous Listening" defined by MIT in 2014 with the advent of voice assistants, we can say that today we are entering "the Era of Ubiquitous Motion Tracking" [1].

Compared to other recognition problems in the domains of computer vision or natural language processing, HAR tasks are relatively easy and do not require extremely powerful computing resources. Several studies have shown that previously trained machine learning models can be run directly on wearable devices without relying on the computing power offered by the cloud [2, 3, 4]. In fact, today's smartwatches are equipped with gigabytes of RAM and adopt multi-core processors capable of providing several GFLOPS of computing power. However, they suffer from a significant limitation due to reduced energy availability, which can significantly reduce battery life when performing complex activities, such as executing machine learning (ML) models that are very energy-intensive.

In general, there are three approaches to implementing HAR on wearable devices: (i) performing the entire computation on the devices; (ii) offloading the entire computation to higher layers; and (iii) partitioning the computation and combining onboard processing and offloading, dynamically choosing where the classification should be made [5, 6, 7]. The last two solutions have to deal with two additional problems: the high energy consumption of the communication interface, which must therefore be used wisely, and the possible loss of privacy due to the transmission and entrustment of sensitive data to a remote entity. From a technical point of view, however, continuous monitoring requires the wearable device to stay in an always-on state for sampling and processing the onboard sensors' data and recognizing human activities.

Typically, HAR applications are interested in a narrow set of activities that need to be monitored. For example, fitness applications focus on particular activities that improve an individual's physical condition, such as running, walking, climbing stairs, and cycling, while ignoring other activities that are part of daily life, such as eating, writing, cooking, and washing. On the other hand, other types of applications with a different purpose might focus on other groups of activities like sanitizing or washing hands ignoring the fitness ones [3]. This means that, given a particular human activity real-

time monitor application, for a good part of the time, the device collects and processes data that contains information that is not of interest, consuming unnecessary energy.

In low-power telecommunication domains, such as wireless sensor networks, this is traditionally known as 'overhearing,' referring to the device being awake to receive and process data not intended for it [8]. In this domain, a solution has been found through the use of remote wake-up triggers, which allow a device to be activated from a low-power state by a signal or command received from a remote source [9, 10, 11]. Unfortunately, in sensor-based HAR the wake-up signal must be searched in real-time within the data coming from the internal sensors. In these conditions, a wake-up could be a particular signature found on the accelerometer or gyroscope data which reveals the presence of one of the activities of interest. If the signature is found the energy-intensive HAR task can be activated to deeply recognize the correct activity. From this point of view, the HAR trigger would behave very similarly to the keyword-spotting technique of modern voice assistants where a hardware sub-system is continuously processing the audio stream to recognize pre-determined keywords or phrases that are used to activate the entire natural language processing (NPL) pipeline [12].

Compared to recognizing a spoken keyword to intentionally activate a device, triggering the HAR task presents several non-trivial challenges. For instance, sometimes it is very difficult, if not impossible, to identify a priori a univocal signature for each activity of interest. Even worse, it may be challenging to find a signature that is common to all the activities of interest and capable of properly activating the recognition task and, at the same time, ignoring the non-interesting activities. Additionally, the trigger recognition must be much less energy-expensive than the complete HAR task to be efficient.

In this work, we propose a novel solution based on a deep-learning lightweight accurate trigger aimed at reducing the energy consumption of the sensor-based continuous HAR. In particular, the contributions of this work can be summarized as follows:

- we provided a problem formulation with a theoretical treatment of the triggering classification performance and energy saving;
- we experimentally characterized on a real wearable device the trade-off between energy consumption and classification accuracy for three types of widely used machine learning models;

- we have shown that the proposed lightweight accurate trigger can reduce energy consumption by up to 95% in realistic case studies at a cost of performance deterioration of at most 1% or 2%.

Notice that, to the best of our knowledge, this is the first contribution in the existing literature describing a lightweight and accurate trigger for machine learning-based human activity recognition in wearable devices.

The remainder of the article is organized as follows: in Section 2 we report the main contributions in the current scientific literature regarding energy-aware human activity recognition; in Section 3 we present the details of the proposed deep-learning-based trigger; in Section 4 we describe the proposed methodology to design a lightweight accurate trigger; in Section 5 we provide a description of the experimental setup used for evaluation of the system (together with the related performance metrics); in Section 6 we illustrate the experiments conducted to characterize the system; in Section 7 we summarize the main contributions and findings of our work.

2. Related work

Recently, executing ML pre-trained models on wearable devices for HAR has become a real thing. Novac et al. in 2020, for instance, compare supervised and unsupervised learning approaches deployed into an embedded device by analyzing the classification accuracy with respect to the ROM footprint and to the inference time [13]; while Chen et al., in 2021, presented a survey on the state-of-the-art of deep learning sensor-based HAR providing information on public data sets that can be used for different tasks [14]. In 2021, Alessandrini et al. presented a recurrent neural network (RNN), deployed on an embedded device, which takes in input data from Photoplethysmography (PPG) and tri-axial accelerometer sensors to infer the current human activity [15]. In 2019, Bhat et al. created custom hardware that integrates all steps of HAR, i.e., reading raw sensor data, feature generation, and activity classification using a deep neural network achieving 95% accuracy and consuming $22.4\mu J$ per operation [16].

Similarly, Coelho et al. [17] and Mayer et al. [18] showed the adequacy of different deep learning models to be run on low-power platforms. Lastly, Wang et al. in [19, 20] work on recognizing human activities from weakly labeled sensor data using Recurrent Attention Networks (RAN) and Attention-Based CNNs showing that these models outperform classical deep learning methods in accuracy and can simultaneously infer multi-activity types.

Although the HAR for low-power devices is the common thread of these last years, in none of the cases do authors analyze trade-offs between network complexity and measured energy consumption in real conditions. On the other hand, partial or total off-loading of the ML task in HAR is relatively brand-new, especially if you take into consideration the works that focus on the problem of energy consumption. In 2020, Samie et al. proposed a hierarchical classification process such that a first classifier executes directly on the IoT device, deciding whether to offload the computation to a gateway or to perform it onboard based on the classification confidence [21]. Similar authors in [22] proposed a two-stage approach that leverages a decision tree (DT) classifier to recognize easy human activity classes, and a 1-dimensional convolutional neural network (1D CNN) to recognize those activities marked as "complex" by the former classifier. A second solution that runs locally on a wearable device and considers also the energy consumed during inference time is that proposed in [23]. Here, the authors proposed an early exit neural architecture search (EExNAS) which, starting from the classification confidence of the model, decides to offload the task to the cloud. Their solution is then benchmarked on a public dataset and the energy consumption is evaluated on an EFM32 processor. In [24] authors proposed an adaptive human activity recognition architecture that consists of a first classier followed by a multilayer CNN. The main goal of this work was to prove that an adaptive architecture, which can dynamically choose at which classification layer exit can result in faster inferencing and save energy. Similarly, Jha et al. in 2022 proposed the Accuracy and Energy-Aware HAR (AE-HAR) model which makes a decision that considers both the energy and the probabilistic accuracy to decide whether to process data locally or not. The idea is to minimize the mobile device energy consumption while maximizing the probabilistic accuracy [25].

Although there are many approaches aimed at reducing the energy consumption of wearable devices in the HAR domain, to the best of our knowledge, no author has explored the use of a triggering system. Triggering has a wide range of applications in signal processing, including data acquisition, waveform generation, time-based analysis, and synchronization of multiple instruments [26]. Several techniques are available to implement the trigger function. One of the conventional approaches is the use of threshold detection, where a trigger signal is detected when the input signal exceeds a certain threshold level [27]. Another approach is the use of pattern recognition algorithms that identify specific patterns or sequences in the input signal. Other

techniques such as correlation-based triggering and edge detection-based triggering are also available [28]. The application of triggering has been extensively used in many signal processing applications, such as in medical signal processing, image processing, and digital signal processing. For instance, in electroencephalography (EEG) signal processing, triggering is useful in detecting specific events such as epileptic spikes or sleep apnea events [29]. In the wireless sensor networks domain, triggering has been used as a solution to reduce the energy consumed to receive and process unintended data [9, 10]. In this context, the use of remote wake-up triggers allows a device to be activated from a low-power state by a command received from a remote source only when an "interesting signal" will be effectively sent to it [11].

In conclusion, in this work, we present a novel solution based on a deep-learning-based lightweight trigger aimed at reducing the energy consumption of wearable devices in the sensor-based HAR domain.

3. The proposed Lightweight Accurate Trigger

In this section, we present the details of the proposed deep-learning-based trigger by presenting the problem formulation, a deep analysis of its energy and classification performances.

3.1. Problem formulation

A sensor-based HAR task is devoted to classifying a set of predefined activities starting from input data such as signals from the accelerometer, gyroscope, magnetometer, etc. Real-time continuous monitoring of human activities entails processing the input stream looking for predefined activities, in order to be able to characterize them, within the possible activities that a person can perform during the day. For instance, a step counter is continuously monitoring the accelerometer signals to find human step signatures to count it. In general, let N denote the number of predefined activities in the classification problem, which are named $a_1, a_2, ..., a_N$, and compose the set A, the HAR problem constructs a function F for predicting the activity sequence starting from the input signals s_t:

$$F(s_t) = \{a_1, a_2, ..., a_t\}, a_t \in A \tag{1}$$

In a real-time continuous recognition system we need to consider that the activities carried out by people are, most likely, more than the ones learned

by any HAR system [30] so that $A \subset M$ with M that represents the set of all possible human activity. With no other gimmick, the unknown activities are matched as any of the available ones, and this leads to misclassifications. Instead, a better approach, presented by Reyes Ortiz et al., introduces the concept of Unknown Activities (UAs). Here, if the prediction confidence level of any known classes is below a certain threshold, the system returns the "unknown" class which means that none of the known activities are present in the input data [31].

Adding the "unknown" class rewrites Equation 1 as:

$$F(s_t) = \begin{cases} \{a_1, a_2, ..., a_t\}, a_t \in A, if\ max(c(a_t)) > th \\ \{unknown\}, otherwise \end{cases} \quad (2)$$

Where $max(c(a_t))$ represents the maxim value of classification confidence of each activity $\in A$.

If on the one hand, this is a solution to the problem of incorrect classification, on the other it still involves a large consumption of computing and energy resources whenever the input does not contain an activity of our interest (i.e. $F(s_t) \notin A$).

Starting from these premises, we propose to make use of a trigger system that starts the classification procedure only when one of the known activities is present in the input signal. Ideally, the trigger system should act as a binary classifier capable of recognizing, with maximum accuracy, the presence or not of one of the interesting classes in the input data, while it should be the most computationally light in order to consume the least possible energy. We call this system Lightweight Accurate Trigger (LAT). In particular, it detects if the input data belongs to the UAs class or to the subset of interesting activities A.

3.2. Energy saving

In general, for an ideal LAT (i.e. a 100% accurate binary classifier), the energy efficiency strictly depends on the ratio between its energy consumption and that of the baseline (i.e. by using only a complete classifier), and on the probability that any of the activities of interest is present in the input signal $p(A)$. The effective classification energy (ECE), starting from the energy consumed by the LAT (E_{LAT}) and by the baseline (E_{BASE}), can be calculated as:

$$ECE = p(A) \times (E_{LAT} + E_{BASE}) + (1 - p(A)) \times E_{LAT} \quad (3)$$

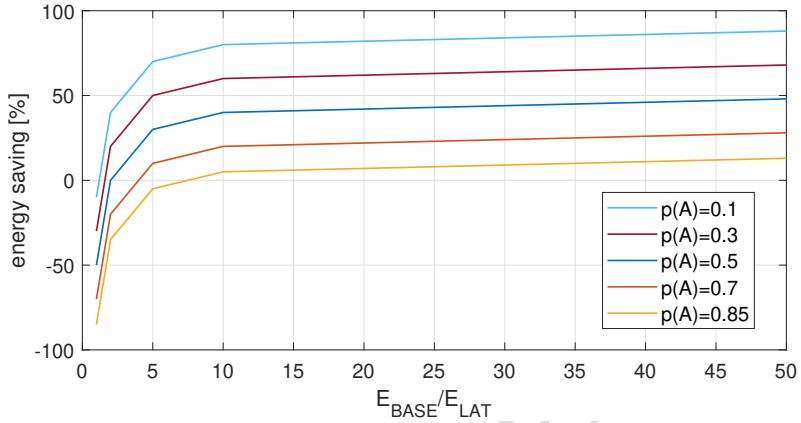


Figure 1: Theoretical energy saved by the triggered approach with respect to the baseline (i.e. by using only the complete classifier) when varying the E_{BASE}/E_{LAT} ratio for different probability $p(A)$

Figure 1 reports the theoretical energy saved by the triggered approach with respect to the baseline when varying the E_{BASE}/E_{LAT} ratio for different probability $p(A)$. As expected, when the probability of finding one of the interesting classes is high, in order to have a significant energy saving, the ratio between the energy consumed by the baseline and by the LAT must be at least one order of magnitude lower. On the contrary, for probability values lower than 0.5, there are large potential gains in terms of energy. For instance, with a LAT that consumes only one-tenth of the baseline energy, we can save up to 55% for probabilities lower than or equal to 0.3.

Dealing with a non-ideal classifier entails considering the energy contributions due to classification errors. In particular, the output of the LAT, regarding the class of the interesting activities A, might be a true positive (TP), true negative (TN), false positive (FP), or false negative (FN). If a TP occurs, the system triggers the complete classifier behaving as in the baseline. In this case, however, the consumed energy, in addition to the basic contribution (E_{BASE}), must also account for the extra energy consumed by the binary classifier E_{LAT}. On the other hand, in the case of TN the total amount of E_{BASE} energy is saved by avoiding executing the complete classifier. When a FP occurs, on the contrary, the system misses the opportunity to save energy by unnecessarily invoking the full classifier and spending an

additional contribution (E_{LAT}) with respect to the baseline. Finally, the FN case reduces the energy consumption but at the cost of misclassification. By taking into account non-idealities, Equation 3 becomes:

$$\begin{aligned} ECE = &\Big[p(A) \times TPR + (1 - p(A)) \times FPR\Big] \times (E_{LAT} + E_{BASE}) + \\ &\Big[(1 - p(A)) \times TNR + p(A) \times FNR\Big] \times E_{LAT} \end{aligned} \tag{4}$$

Where TPR, FPR, TNR, and FNR represent, respectively, the rate of the corresponding true or false output.

Considering that, for a given classifier:

$$TPR = 1 - FNR \tag{5}$$

and

$$TNR = 1 - FPR \tag{6}$$

we can rewrite Equation 4 using only FPR and FNR:

$$\begin{aligned} ECE = &\Big[p(A) \times (1 - FNR) + (1 - p(A)) \times FPR\Big] \times (E_{LAT} + E_{BASE}) + \\ &\Big[(1 - p(A)) \times (1 - FPR) + p(A) \times FNR\Big] \times E_{LAT} \end{aligned} \tag{7}$$

Figure 2 shows the theoretical energy saved with respect to the baseline of the proposed approach when varying the FPR of the LAT for different probabilities $p(A)$. Since each FP results in a missed opportunity to save energy, it is obvious that as the FPR increases, the percentage of energy saved by the system decreases. Moreover, the energy waste results higher for lower values of $p(A)$ which, in turn, means a higher number of negative samples in the input which proportionally increases the number of FP.

On the contrary, Figure 3 shows how the percentage of energy saved by the system depends on the FNR for different values of the probability $p(A)$. In particular, for all reported values of probability, there is an increase in the amount of energy saved which, in turn, is greater for higher values of $p(A)$. Although there is a decrease in energy consumption, it must not be forgotten that this is not a viable way to reduce it, as each FN corresponds to an unrecoverable failure of the classification system.

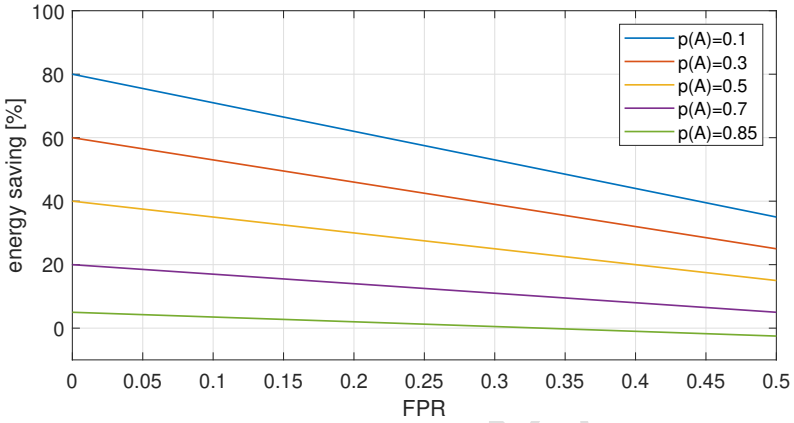


Figure 2: Theoretical energy saved by the triggered approach with respect to the baseline when varying the FPR of the LAT classifier for different values of the probability $p(A)$

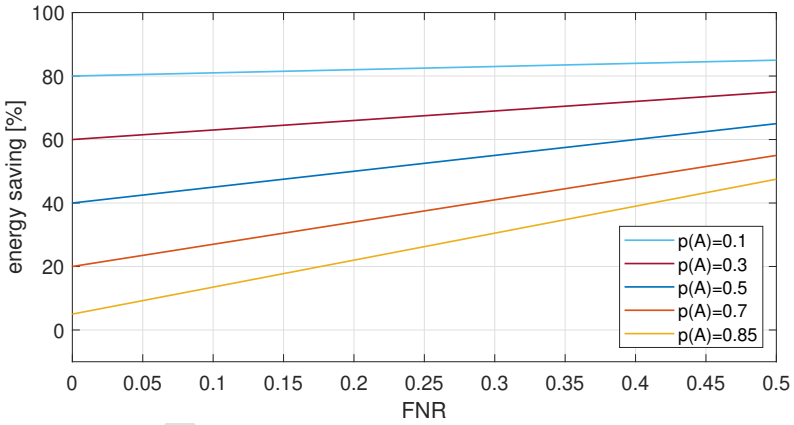


Figure 3: Theoretical energy saved by the triggered approach with respect to the baseline when varying the FNR of the LAT classifier for different values of the probability $p(A)$

In summary, for the proposed approach in order to produce significant energy savings, the following conditions are simultaneously required:

$$E_{BASE} \gg E_{LAT} \tag{8}$$

$$p(A) \ll 1 \tag{9}$$

$$FPR = as\ low\ as\ possible \tag{10}$$

3.3. Classification performances

One of the constraints to realizing a good LAT is to keep the overall misclassification rate at least as low as the baseline. In the proposed approach, misclassification happens in one of the following cases:

- the LAT misclassifies a true input as false (FN) so that the complete classifier is not triggered;
- the LAT correctly identifies a positive input (TP), the complete classifier is triggered, but it misclassifies the input.
- the LAT misclassifies a false input (FP), the complete classifier is triggered, but it misclassifies the input.

Definitely, the misclassification rate of the proposed system MCR_{system} can be described by:

$$MCR_{system} = FNR + TPR \times MCR_{base} + FPR \times MCR_{base} \tag{11}$$

Where FNR, TPR, and FPR refer to LAT and MCR_{base} is the misclassification rate of the baseline. Moreover, by rewriting Equation 11 as

$$MCR_{system} = FNR + (TPR + FPR) \times MCR_{base} \tag{12}$$

we can highlight that the misclassification rate of the proposed system (MCR_{system}) is dominated primarily by the value of FNR because each TP sample returned by the LAT involves the normal activity of the base classifier and each FP should largely be mitigated by its higher classifying capability. For these reasons, the last requirement that a LAT should meet to be suitably applied to a continuous HAR system is:

$$FNR = as\ low\ as\ possible \tag{13}$$

Notice that, concerning the four mentioned requirements, it must be specified that, as (10) and (13) deal with the classification performance, the LAT architecture should be chosen properly in order to meet these constraints. In the same way, the fulfillment of requirement (9) strictly depends on the architecture complexity of the LAT which translates into its energy consumption. Finally, equation (9) suggests that the LAT cannot be applied to those applications in which the probability of occurrence of interesting activities is too high.

4. Design methodology

A LAT for sensor-based HAR can be designed in two different ways: (i) through traditional signal processing and template matching techniques, or (ii) via a holistic approach based on machine learning. The first approach requires a deep understanding of the physical model of each human activity in order to identify the most appropriate signatures that allow for rapid recognition. On the other hand, creating a trigger sensitive to multiple activities seems far from a simple possibility. Instead, our proposed approach plans to train a binary classifier based on machine learning techniques that can recognize the presence of a set of interesting activities within the signals from the accelerometer and gyroscope.

In this work, we proposed to use both convolutional neural networks (CNNs) and Long Short Term Memory (LSTM) networks thanks to its improved accuracy and effectiveness established in HAR tasks [32, 33]. Furthermore, studies have shown that combining CNN and LSTM networks provides a promising solution for HAR classification problems. Various studies have proposed hybrid CNN-LSTM architectures, such as DeepConvLSTM and AttSense, which have proven to outperform pure LSTM networks [34, 35]. The reason behind such a successful approach is that the combined adoption of CNN and LSTM architecture allows for capturing both spatial and temporal features, which are two characteristics that are typical of human activities extracted from sensor devices. Therefore we evaluated the performance of all three types of deep networks both to create the LAT and the complete classifier.

4.1. The n+1 dataset for HAR

There are many datasets traditionally used to train and test machine learning models in HAR but, unfortunately, almost none contain also data related to *UAs* sampled during normal daily life. We define the latter as *n+1* datasets since in addition to the n different classes used in HAR problems they contain one more class that represents data collected in the most varied human activities which do not include any of the n previous.

In the following, we provide a procedure to derive both a *n+1* and a *binarized* dataset starting from a classic HAR dataset to build, respectively, a traditional classifier and a LAT. In particular, given A the entire set of activities stored in a traditional dataset, we can obtain a corresponding *n+1*

($\widetilde{A}$) and a binarized ($\widehat{A}$) dataset by creating two partitions A' and A'' containing, respectively, the records relating only to some of the activities of the original dataset, which we chose as activity of interest, and the rest we label as unknown activities (UAs). To obtain $\widetilde{A}$ we rewrite the original labels of the A'' partition as "0" while for the $\widehat{A}$ we rewrite also the labels of the A' partition as "1".

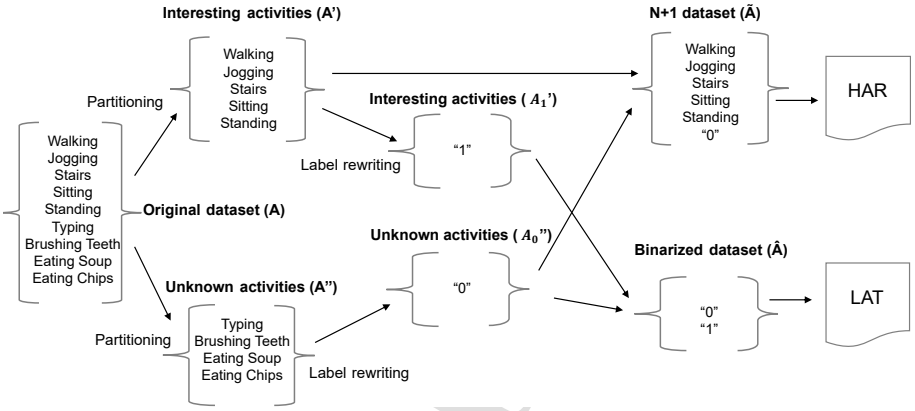


Figure 4: The proposed workflow to train and test both LAT and HAR models starting from common HAR datasets.

Figure 4 shows an example of the application of the proposed procedure to a particular dataset containing the following activities: *Walking, Jogging, Stairs, Sitting, Standing, Typing, Brushing Teeth, Eating Soup, and Eating Chips*. In the first step, from the original dataset, we derive A' containing the activities: *Walking, Jogging, Stairs, Sitting, and Standing* and A'' containing *Typing, Brushing Teeth, Eating Soup, and Eating Chips*. Each label of the records contained in the A'' partition is then replaced with "0" to obtain a single label partition A''_0. In the same way, the single label partition A'_1 is obtained by replacing with "1" each label of A'. Finally, a binarized dataset $\widehat{A}$ is obtained by combining A''_0 and A'_1 which is then used to explore the design space of the LAT. In parallel, the *n+1 dataset* ($\widetilde{A}$) is obtained by adding the unknown activities (the A''_0 partition) to the A' partition (i.e. the activities of interest). The $\widetilde{A}$ is then used to tune the design space of a traditional HAR system. Notice that, to maintain a balanced number of samples among the classes, after grouping different activities, a rebalancing phase is

needed, for example, by means of random undersampling.

The choice of how to partition the dataset introduces an interesting parameter for which to evaluate sensitivity because it is presumable that different combinations of activities lead to different performances of the LAT. For example, it seems intuitive that considering activities of interest that are rather similar but very different from what we chose as *UAs*, the binary classifier behaves better than in the reverse case. This will therefore be one of the objects of study of the present work.

4.2. Design workflow

As described in section 3, the efficiency and the effectiveness of a LAT should be defined with respect to a traditional approach in which we used an always-on classifier. For this reason, to design a LAT we must first build and characterize the traditional classifier. Figure 5 shows the proposed workflow used to build both the classifier and to evaluate the best LAT configuration taking into account the trade-off between energy saved and classification accuracy. In particular, starting from the original dataset, we create the binarized ($\widehat{A}$) and the *n+1 datasets* ($\widetilde{A}$) as described in the previous section, then we start building the classification model (the right part of the figure). Notice that, in this work, the classifiers focus on the signals gathered from the triaxial accelerometer and gyroscope that are properly divided into time windows lasting 2 seconds with a 75% overlap. Choosing the size of the time window and the percentage of overlap is a non-trivial task because the length of the time window impacts the classification performance of the models [36]. For instance, in HAR tasks, different window lengths have been used in the literature, from 1 s up to 30 s [4, 37, 38, 39]. In this study, we opted for a 2 seconds time window and for 75% of overlap as a reasonable choice, given the characteristics of the adopted datasets, in terms of activities to be classified and of sampling ratios, as reported also in[3, 4]. Then, the windowed data are used to find the best hyperparameters setting of the classifier by means of a Hyperband Tuner that takes in input the dataset, split into training, validation, and testing, together with an exploration range for each model hyperparameter[40, 41].

Once the best configuration of the hyperparameters is found, the models are trained and tested using the original dataset divided into "train" and "test", where 75% are used as the training set and 25% as the test set. The training-testing procedure is performed five times with five different seeds of the random generator in order to have five different dataset splits to evaluate

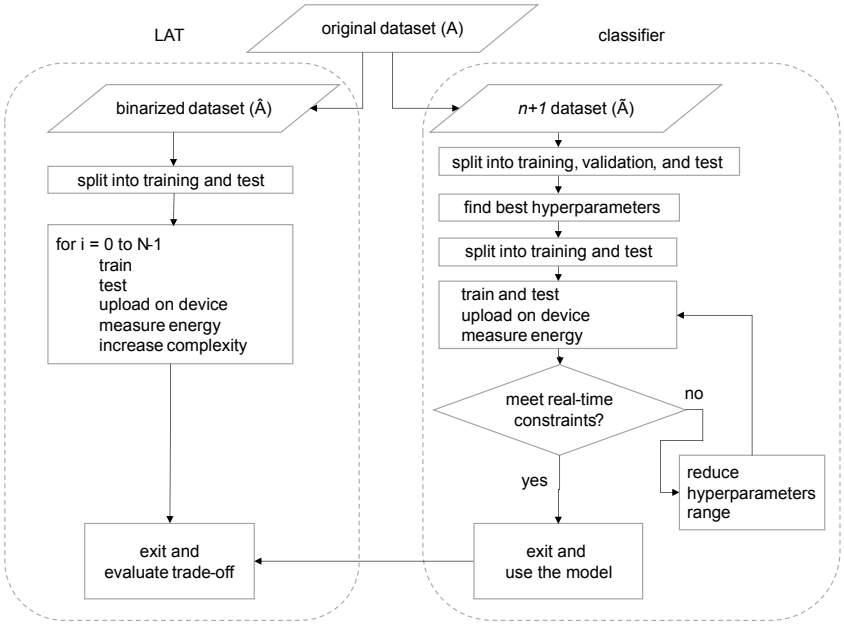


Figure 5: The proposed workflow to build both the LATs and the complete models.

the robustness of the approach. Then, each model, after being trained and tested, is uploaded to a wearable device to evaluate its energy consumption and inference time.

Since the classifier will have to carry out the inference in real-time, it is necessary that it respects the time constraints foreseen in the design phase. In particular, since the sensor-based HAR is carried out over time windows of a predetermined size (2 seconds in this work), the time required to carry out the inference must never exceed the length of this window. If not, it will be necessary to revise the model by reducing the range of hyperparameters in the Hyperband Tuner and restarting characterizing it.

Once the best classification model has been found, we proceed to the design and evaluation of different LAT models with increasing complexity using the dataset $\widehat{A}$ (left side of Figure 5). Notice that, in this case, we don't make use of an automatic hyperparameters tuner, because we don't just look for the model that performs better, but we characterize a family of

LAT models with increasing complexity in order to be able to evaluate the possible trade-off between accuracy and energy consumption.

5. Experimental setup

In this section, we present the experimental setup provided to evaluate the effectiveness and energy efficiency of the proposed approach.

5.1. Machine learning models

To build the baseline classifier we proposed to tune the following models: i) an LSTM network composed of three LSTM layers followed by three dense layers (3LSTM_3D); ii) a CNN network composed of three convolutional layers followed by three dense layers (3Conv_3D); a hybrid network composed of two convolutional layers and two LSTM layers followed by three dense layers (2Conv_2LSTM_3D); Notice that, in order to reduce the overfitting, we intersperse each dense layer with a dropout layer with a drop probability set to 0.2.

Concerning the LAT, we explored the hyperparameters space of the following networks obtained by reducing the deepness of the previous one: i) an LSTM network composed of two LSTM layers followed by a single dense layer (2LSTM_1D); ii) a CNN network composed of two convolutional layers followed by a single dense layer (2Conv_1D); iii) a hybrid network composed of one convolutional layer and one LSTM layer followed by a single dense layer (1Conv_1LSTM_1D). Also in this case we add a dropout layer to reduce overfitting.

To implement and test the proposed models we make use of the TensorFlow end-to-end machine learning platform powered by Google [42] and to find the best hyperparameter configuration of each classifier, as reported in Section 4.2, we make use of the Hyperband Tuner provided by Keras [40, 41]. Once trained and characterized, each model has been installed and executed on a real smartwatch using the TensorFlow Lite library [43]. All phases of training, testing, and exploration of the hyperparameter space were conducted on a workstation equipped with two Intel® Xeon® Silver 4314 processors and three NVIDIA® A100 graphical processing units (GPU).

5.2. Datasets

We evaluated the proposed approach on two publicly available HAR datasets, namely *WISDM* and *Watch_HAR*, and on a specially created dataset we called *Ad-hoc DB*.

WISDM [44]: this dataset, also known as "WISDM Smartphone and Smartwatch Activity and Biometrics Dataset" was published by Gary Weiss in 2019 and it includes data collected from 51 subjects, each of whom was asked to perform 18 tasks for three minutes each. Data (sampled at 20 Hz) contains both accelerometer and gyroscope signals and were recorded using a smartwatch placed on the dominant hand and a smartphone placed on the pocket during the following activities: *walking, jogging, stairs, sitting, standing, typing, brushing teeth, eating soup, eating chips, eating pasta, drinking, eating a sandwich, kicking a soccer ball, playing tennis, dribbling, writing, clapping, and folding clothes.*

Watch_HAR [45]: published by Ada Alevizaki and Niki Trigoniit in 2022, contains data collected in a laboratory environment, where users performed activities commonly executed in the home or work environments while wearing a smartwatch on their dominant hand. The 13 users, both male and female, aged between 23 and 67 years old, executed in their own style the following 16 activities: *brushing teeth, preparing sandwich, reading book, typing, using phone, using remote control, walking freely, walking holding a tray, walking with handbag, walking with hands in pockets, walking with object underarm, washing face and hands, washing mug, washing plate, writing.* Each user performed each activity for approximately 1 to 3 minutes.

Ad-hoc DB: we built an *n+1* HAR dataset aimed at studying the recognition of *handwashing* and *handrubbing* activities performed during the day. In particular, we collected sensor data from the triaxial accelerometer and gyroscope of a smartwatch positioned on the wrist of the dominant hand of four participants during real-life activities. Each subject was wearing the smartwatch for several hours on different days and was asked to annotate the start and the end of each handwashing or handrubbing activity performed during the day. Together with the interesting activities, we collected also *UAs* data by randomly sampling the sensors during the day. For each subject, we collected about 2 hours of total time spent washing hands, about 2 hours and 30 minutes of time spent rubbing, and about 3 hours of *UAs*. Notice that the subjects were not instructed on how to wash or rub their hands leaving them completely free to use their usual way so to collect data about the unstructured way people normally use to wash their hands.

5.3. The wearable device

As a case study, we used an OPPO Watch 46mm equipped with a Qualcomm ® Snapdragon Wear™ 3100 which has 1 GB of ram and 8 GB of

non-volatile flash memory. The smartwatch is also equipped with several sensors such as accelerometer, gyroscope, magnetic field sensor, barometer, etc. The whole device is powered by means of a 430 mAh Li-Po battery that ensures up to 36 hours of autonomy. Concerning the software, the device runs Wear OS by Google which allows a high degree of programmability and also supports the TensorflowLite libraries for machine learning. For the purpose of this work, the smartwatch has been opened, the battery has been removed and it has been powered using measurement setup described below.

5.4. Energy Consumption Measurement Setup

In order to monitor the energy consumption, the smartwatch has been opened as shown in Figure 6 and, after removing the battery, we measured the voltage drop across a sensing resistor (1.5Ω) placed in series with the power supply of the device. The smartwatch was powered at 3.85V through an NGMO2 Rohde & Schwarz dual-channel power supply [46], and we sampled the signals during the experiments by means of a National Instruments NI-DAQmx PCI-6251 16-channel data acquisition board connected to a BNC-2120 shielded connector block [47, 48].

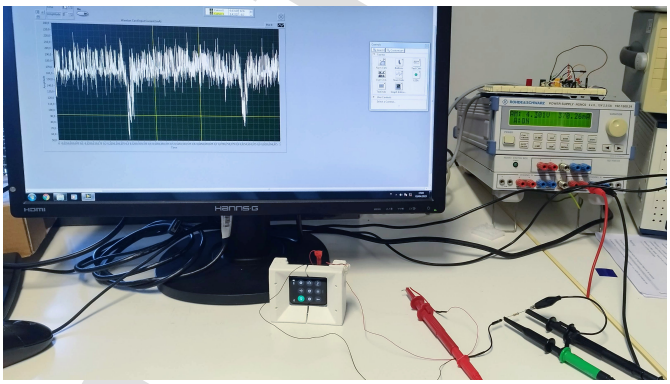


Figure 6: The OPPO Watch 46mm connected to the external power supply and to the measurement setup.

5.5. Classification Performance Metrics

To evaluate the performance in the classification of multi-class problems, the following quantities have been evaluated for each of the classes of the

datasets ($i \in [1 \cdots N]$ is an index that identifies a specific class): TP_i, the number of true positives predicted for class i; TN_i, the number of true negatives predicted for class i; FP_i, the number of false positives predicted for class i; FN_i, the number of false negatives predicted for class i.

Subsequently, these indicators have been used to compute the following metrics (corresponding to the so-called *macro-averaging* measures) [49]:

$$Precision = \frac{1}{N}\sum_{i=1}^{N}\frac{TP_i}{TP_i + FP_i} \tag{14}$$

$$Recall = \frac{1}{N}\sum_{i=1}^{N}\frac{TP_i}{TP_i + FN_i} \tag{15}$$

$$Accuracy = \frac{1}{N}\sum_{i=1}^{N}\frac{TP_i + TN_i}{TP_i + TN_i + FP_i + FN_i} \tag{16}$$

6. Experiments and results

In the following, we present the results of the analysis performed to characterize the effectiveness and efficiency of the proposed approach.

6.1. Performance of the classifications models

Table 1 shows the results of the hyperparameters exploration for each of the proposed models. In particular, H1, H2, H3, and H4 represent the number of the hidden units inside of a layer, for instance, in a convolutional layer they are the number of internal filters while in an LSTM layer are the number of cell units. On the other hand, D1, D2, and D3 represent the number of computing elements (neurons) inside each dense layer.

The table also reports the size of each model once converted to TensorFlowLite format and the corresponding average execution time and energy consumption, together with the standard deviation, measured in the inference phase repeated ten times. Both 3LSTM_3D and 2Conv_2LSTM_3D networks required several iterations forcing the hyperparameter space reduction to be able to respect the real-time constraints while, concerning the 3Conv_3D network, the best configuration found by the tuner was directly matching the requirements. Due to the high efficiency of the simple convolutional network,

Table 1: Hyperparameter setting of the proposed machine learning classifiers together with the model size, inference time, and energy consumption.

parameter	**3LSTM_3D**	**3Conv_3D**	**2Conv_2LSTM_3D**
H1	544	192	320
H2	512	864	1024
H3	896	928	1024
H4	-	-	608
D1	416	416	512
D2	576	224	192
D3	#classes	#classes	#classes
size [kB]	10,101	10,137	10,000
time [ms]	$1,987 \pm 96$	681 ± 23	$1,808 \pm 88$
energy [mJ]	306 ± 12	123 ± 5	252 ± 11

the measured inference time is around 681 ms which results in energy consumption of just 123 mJ while the other two models show inference times close to two seconds with consequent energy consumption which reaches up to 306 mJ.

From the classification performance point of view, the 3Conv_3D network outclasses the other two in all three datasets. Table 2 reports the performances obtained by the proposed models using the hyperparameter settings described in Table 1. Notice that, each value, together with its standard deviation, has been obtained as the average of five training-testing runs using different random seeds to split the dataset into training and testing partitions. The 3Conv_3D network reaches the best accuracy of about 96% on top of the WISDM dataset while 3LSTM_3D and 2Conv_2LSTM_3D achieved a maximum accuracy of 89% and 91%, respectively. Even referring to precision and recall, the 3Conv_3D network is always the best with a gap ranging from about 8% to 3%. The low value of the standard deviations measured for each model testifies to the substantial independence of the results from the dataset-splitting procedure.

On the other hand, in these experiments, Watch_HAR and WISDM have been converted into *n+1* datasets by choosing the partitioning that makes more sense, i.e. trying to form the two groups by relegating in the first one the activities that make greater use of arms and hands and in another, the activities that involve the whole body more. For instance, concerning the

Table 2: Best classification performances obtained on top of the three reference datasets

metric	dataset	3LSTM_3D	3Conv_3D	2Conv_2LSTM_3D
accuracy	Ad-hoc DB	0.878 ± 0.008	0.916 ± 0.004	0.886 ± 0.010
	Watch_HAR	0.821 ± 0.008	0.916 ± 0.009	0.824 ± 0.007
	WISDM	0.889 ± 0.006	0.958 ± 0.002	0.909 ± 0.008
precision	Ad-hoc DB	0.882 ± 0.006	0.915 ± 0.004	0.889 ± 0.003
	Watch_HAR	0.822 ± 0.004	0.908 ± 0.006	0.825 ± 0.007
	WISDM	0.887 ± 0.005	0.942 ± 0.005	0.909 ± 0.009
recall	Ad-hoc DB	0.881 ± 0.007	0.915 ± 0.003	0.894 ± 0.010
	Watch_HAR	0.817 ± 0.009	0.911 ± 0.014	0.823 ± 0.008
	WISDM	$0.887 \pm 0,007$	0.978 ± 0.003	0.907 ± 0.005
MCR	Ad-hoc DB	0.122 ± 0.008	0.084 ± 0.004	0.114 ± 0.010
	Watch_HAR	0.179 ± 0.008	0.084 ± 0.009	0.176 ± 0.007
	WISDM	0.111 ± 0.006	0.042 ± 0.002	0.091 ± 0.008

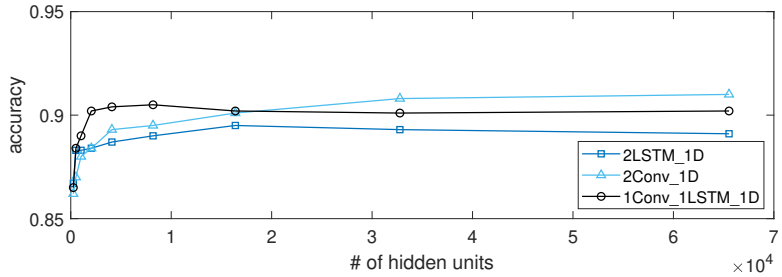
Whatch_HAR dataset, we have chosen as an activity of interest: *brushing teeth, preparing sandwich, reading book, typing, using phone, using remote control, washing face and hands, washing mug, washing plate, writing* while the following activities have been chosen as *UAs*: *walking freely, walking holding a tray, walking with handbag, walking with hands in pockets, walking with object underarm.*

Similarly, regarding the WISDM dataset, we have chosen the following as activities of interest: *typing, brushing teeth, eating soup, eating chips, eating pasta, drinking, eating a sandwich, writing, clapping, and folding clothes* while as uninteresting activities we chose: *walking, jogging, stairs, sitting, standing, kicking a soccer ball, playing tennis, dribbling.* Notice that, in Section 6.3 we will show the results of a deep analysis of the sensitivity of the proposed approach concerning the choice of partitioning.

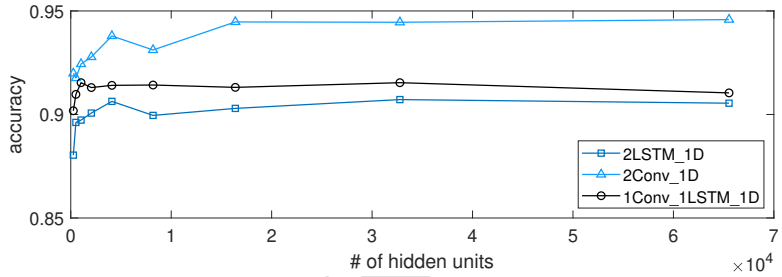
Given its enormous energy efficiency and high classification capability, the 3Conv_3D network was used as a reference base model in the rest of the paper.

6.2. LAT characterization

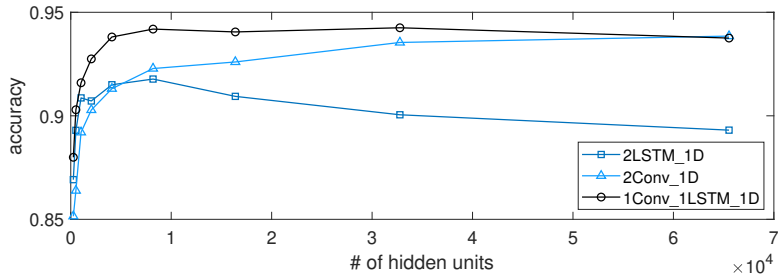
Concerning the LAT characterization, we explored the hyperparameters space of the following three proposed network models on top of the three



(a) Ad-hoc DB



(b) Watch_HAR

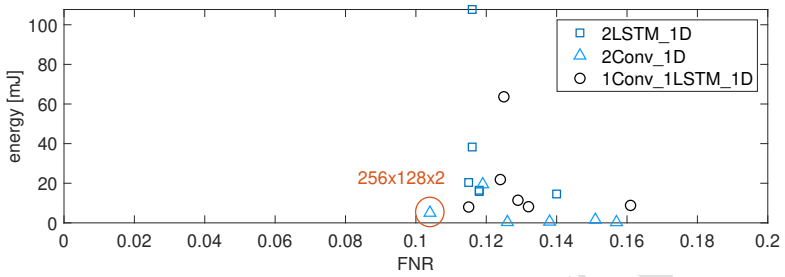


(c) WISDM

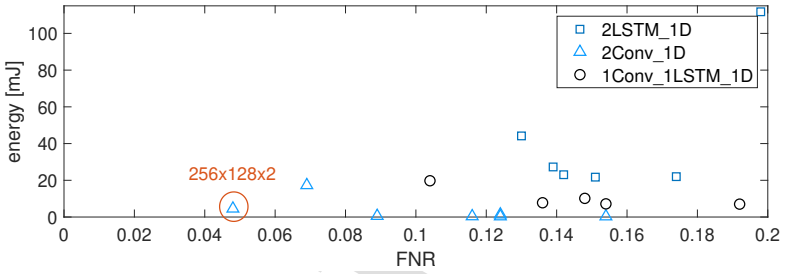
Figure 7: Accuracy obtained by the proposed LAT models when varying the total number of hidden units on top of the three selected datasets.

binarized datasets.

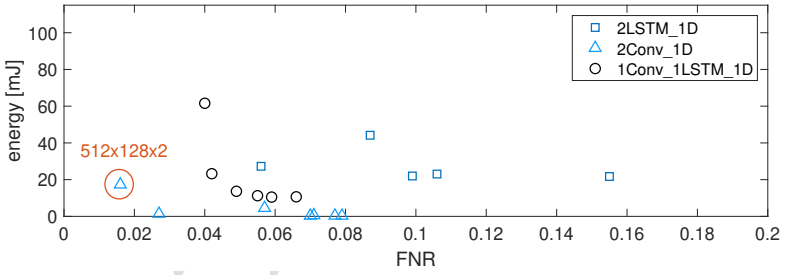
Figure 7 reports the classification accuracy obtained by the LAT models when varying the total number of hidden units on top of the three selected datasets. As expected, increasing the network size increases its classification



(a) Ad-hoc DB



(b) Watch_HAR



(c) WISDM

Figure 8: Pareto diagram reporting the energy consumed in a single inference by each LAT configuration Vs its *FNR*.

accuracy which, however, after a certain level reaches the maximum value and then stabilizes or, in the case of 2LSTM_1D on WISDM, starts to decrease significantly. Moreover, as for the baseline classifier, the pure convolutional model (2Conv_1D) produces the best performance in terms of classification

capability although in Ad-hoc DB and in WISDM it is equated by the hybrid configuration (1Conv_1LSTM_1D).

Figure 8 shows the Pareto diagrams obtained by plotting the energy consumed in a single inference by each LAT configuration versus the corresponding *FNR*. On each plot, the configuration that represents the best compromise between energy consumption and *FNR* level (i.e. the point minimizing both energy consumption and FNR) has been circled in red. For instance, concerning the Ad-hoc DB and Watch_HAR datesets, the best model turns out to be the 2Conv_1D with 256 and 128 hidden units respectively in the first and second layers. Similarly, in the WISDM dataset, the best model is still a 2Conv_1D network but with 512 and 128 hidden units.

Therefore, thanks to its high energy efficiency and high classification accuracy, the two-layer convolutional network turns out to be the best deputy to build a LAT. For this purpose, we reported in Table 3 the complete characterization of the proposed system assuming the 3Conv_3D network as the base classifier to which the LAT, consisting of the 2Conv_1D model with the best configuration, was applied. The table reports, for the three datasets, the average classification performance ad its standard deviations together with the LAT size, inference time, and energy obtained in five runs using different random seeds. From the classification performance point of view, as described in Section 3.3, we reported the *FNR, TPR, FPR*, MCR_{base}, MCR_{system}, $MMCR_{system}$, and the $\Delta MMCR_{system}$ where the last two metrics refer, respectively, to the measured MCR_{system} and the delta $MMCR_{system}$. The latter, in particular, is the misclassification overhead introduced by the LAT with respect to the base classifier. Notice that, while the MCR_{system}, which has been calculated using Equation 12, represents the theoretical misclassification, the $MMCR_{system}$ has been measured in the experimental conditions by cascading the trained LAT and the base classifier and by testing the resulting chain on top of the three available datasets. The reason why the theoretical and the measured MCR can be significantly different is to be found in the fact that the samples poorly classified by the LAT (generating FN, for example) are largely the same ones for which the baseline classifier makes errors. Consequently, in several cases, the errors made by the two classifiers in cascade do not add up, effectively reducing the real measured MCR of the system.

Concerning the results shown in Table 3, it can be seen that the *FNR* of the LAT varies from 10% to about 2% depending on the dataset used. In the worst case, i.e. using the Ad-hoc DB, the theoretical MCR_{system} changes

Table 3: Best performance of the proposed LAT model together with the model size, inference time, and energy consumption.

	Ad-hoc DB	**Watch_HAR**	**WISDM**
# of elements	256x128x2	256x128x2	512x128x2
FNR	0.104 ± 0.035	0.048 ± 0.011	0.016 ± 0.003
TPR	0.896 ± 0.035	0.952 ± 0.011	0.984 ± 0.003
FPR	0.102 ± 0.019	0.071 ± 0.014	0.027 ± 0.009
MCR_{base}	$\mathbf{0.084 \pm 0.004}$	$\mathbf{0.084 \pm 0.009}$	$\mathbf{0.042 \pm 0.002}$
MCR_{system}	0.188 ± 0.041	0.134 ± 0.029	0.059 ± 0.020
$MMCR_{system}$	$\mathbf{0.095 \pm 0.013}$	$\mathbf{0.091 \pm 0.009}$	$\mathbf{0.044 \pm 0.003}$
$\Delta MMCR_{system}$	$\mathbf{0.011 \pm 0.017}$	$\mathbf{0.007 \pm 0.018}$	$\mathbf{0.002 \pm 0.005}$
base classifier (3Conv_3D)			
size [kB]		10,137	
time [ms]		681 ± 23	
energy [mJ]		123 ± 5	
LAT			
size [kB]	423	423	808
time [ms]	58.38 ± 3.32	56.32 ± 5.61	213.60 ± 11.06
energy [mJ]	4.73 ± 0.94	4.57 ± 0.81	17.32 ± 6.30
E_{base}/E_{LAT}	$\mathbf{26.01 \pm 6.23}$	$\mathbf{25.89 \pm 5.87}$	$\mathbf{7.10 \pm 2.87}$

from 0.084, obtained by the base classifier (MCR_{base}), to 0.188 introducing an overhead of more than 10 percentage points even if, fortunately, the measured misclassification rate ($MMCR_{system}$) does not exceed 0.095. In this way, the real overhead ($\Delta MMCR_{system}$) is limited to about $1\% \pm 1.7\%$. In the case of Watch_HAR and WISDM datasets, the results appear even better with a real overhead of around $0.7\% \pm 1.8\%$ and $0.2\% \pm 0.5\%$ respectively. Therefore, it can be concluded that, from the experimental data, the introduction of the proposed LAT leads to a performance deterioration of at most 1% or 2% compared to the basic classifier.

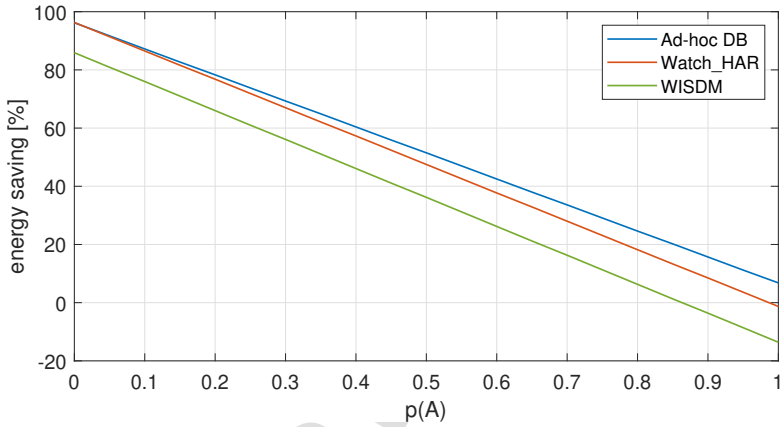


Figure 9: Theoretical energy saved by the triggered approach with respect to the baseline when varying the FNR of the LAT classifier for different values of the probability $p(A)$

Concerning the resource utilization (energy, time inference, and energy), the table reports for comparison also the characterization data of the base classifier (3Conv_3D)) to which the LAT was applied. Thanks to the low complexity of the LAT and to its reduced inference time, results that the energy ratio between the base classifier and the LAT ranges from 7× to 26× depending on the number of hidden units of the LAT. Likewise, the size of the LAT models ranges from about 400 to 800 kB whereas the base classifier occupies more than 10 MB of memory.

Figure 9 plots the energy saved by the proposed approach when varying $p(A)$, i.e. the probability that any of the activities of interest is present in the input signal, by calculating ECE as reported in Equation 7, starting from

the measured values reported in Table 3. In particular, these results show, for instance, that for a 0.5 probability of having one of the activities of interest of the WISDM $n+1$ dataset, the proposed approach can save almost 40% of the energy. In the case of the Watch_HAR and Ad-hoc DB, on the other hand, energy saving exceeds 50%. Moreover, considering a realistic case, i.e. a person wearing a smartwatch on which an application for monitoring some of the daily activities is installed, this percentage could grow further. Take as an example the application described in [3] which has the task of monitoring the washing and sanitization of hands (Ad-hoc DB). For this application, it can be estimated that if the person washed/sanitized their hands 30 times a day and each time took 30 seconds, $p(a) = 30 \times 30/86400 = 0.01$ giving rise to an energy saving of about 95%. On the other hand, concerning the WISDM dataset, we can imagine, for example, a person who spends about 6 hours a day, $p(a) = 6/24 = 0.25$, in *typing, brushing teeth, eating soup, eating chips, eating pasta, drinking, eating a sandwich, writing, clapping, and folding clothes* and, in this case, the proposed approach will save up to 60% of smartwatch energy.

6.3. Sensitivity to the clustering of the activities

To investigate how much the choice of which activities to include in the A' and A'' partitions of the derived datasets affects the performance of the proposed approach, we performed a series of experiments in which human activities were randomly assigned to the two partitions, and on these, the whole system has been characterized. The workflow has been repeated ten times using different random seeds in order to measure the average performance together with the standard deviation.

Table 4: Classification performance of the LAT and the base classifier when randomly choosing the composition of partitions A' and A'' for the WISDM dataset.

	accuracy	precision	recall
Classifier	0.9425 ± 0.0061	0.9413 ± 0.0062	0.9307 ± 0.0065
LAT	0.9559 ± 0.0115	0.9509 ± 0.0145	0.9506 ± 0.0124

Table 4 reports the results obtained by training and testing both the best base classifier and the LAT on top of the WISDM dataset. In both cases, the reduced value of the measured standard deviations suggests a negligible dependence on the activities selected to be assigned to the two partitions

contrary to what initially seemed presumable, ultimately confirming the robustness of the proposed approach.

7. Conclusion

Continuous HAR has become increasingly popular, thanks to smartphones and smartwatches that are equipped with various sensors, such as accelerometers, gyroscopes, magnetometers, and heart rate monitors. These sensors enable constant activity recognition. In the era of ubiquitous motion tracking, the issue of energy consumption in battery-powered wearable devices has become more pressing, as it limits the usage time.

In this paper, we propose a lightweight and accurate trigger to reduce energy consumption in sensor-based continuous HAR using deep learning models. We provide a theoretical formulation of the proposed approach and a comprehensive experimental characterization of its energy consumption, measured on a real hardware wearable device. The experimental results, conducted on realistic scenarios, showed energy savings ranging from 60% to 95%, depending on the selected HAR application, with a performance deterioration cost of at most 1% or 2% compared to the basic classifier.

References

[1] D. Talbot, The Era of Ubiquitous Listening Dawns, Tech. rep., MIT Technology Review (2013) [cited 10 March 2023].
URL `https://www.technologyreview.com/2013/08/08/177068/the-era-of-ubiquitous-listening-dawns/`

[2] X. Wang, M. Magno, L. Cavigelli, L. Benini, Fann-on-mcu: An open-source toolkit for energy-efficient neural network inference at the edge of the internet of things, IEEE Internet of Things Journal 7 (5) (2020) 4403–4417. doi:10.1109/JIOT.2020.2976702.

[3] E. Lattanzi, L. Calisti, V. Freschi, Unstructured handwashing recognition using smartwatch to reduce contact transmission of pathogens, IEEE Access 10 (2022) 83111–83124. doi:10.1109/ACCESS.2022.3197279.

[4] E. Lattanzi, M. Donati, V. Freschi, Exploring artificial neural networks efficiency in tiny wearable devices for human activity recognition, Sensors 22 (7) (2022) 2637.

[5] S. M. Tahsien, H. Karimipour, P. Spachos, Machine learning based solutions for security of internet of things (iot): A survey, Journal of Network and Computer Applications 161 (2020) 102630.

[6] F. Samie, L. Bauer, J. Henkel, From cloud down to things: An overview of machine learning in internet of things, IEEE Internet of Things Journal 6 (3) (2019) 4921–4934.

[7] M. Xu, F. Qian, M. Zhu, F. Huang, S. Pushp, X. Liu, Deepwear: Adaptive local offloading for on-wearable deep learning, IEEE Transactions on Mobile Computing 19 (2) (2019) 314–330.

[8] P. Basu, J. Redi, Effect of overhearing transmissions on energy efficiency in dense sensor networks, in: Proceedings of the 3rd international symposium on Information processing in sensor networks, 2004, pp. 196–204.

[9] A. Bogliolo, E. Lattanzi, V. Freschi, Idleness as a resource in energy-neutral wsns, in: Proceedings of the 1st International Workshop on Energy Neutral Sensing Systems, 2013, pp. 1–6.

[10] U. Raza, A. Bogliolo, V. Freschi, E. Lattanzi, A. L. Murphy, A two-prong approach to energy-efficient wsns: Wake-up receivers plus dedicated, model-based sensing, Ad Hoc Networks 45 (2016) 1–12.

[11] E. Zaraket, N. M. Murad, S. S. Yazdani, L. Rajaoarisoa, B. Ravelo, An overview on low energy wake-up radio technology: Active and passive circuits associated with mac and routing protocols, Journal of Network and Computer Applications 190 (2021) 103140.

[12] I. López-Espejo, Z.-H. Tan, J. H. Hansen, J. Jensen, Deep spoken keyword spotting: An overview, IEEE Access 10 (2021) 4169–4199.

[13] P.-E. Novac, A. Castagnetti, A. Russo, B. Miramond, A. Pegatoquet, F. Verdier, Toward unsupervised human activity recognition on microcontroller units, in: 2020 23rd Euromicro Conference on Digital System Design (DSD), IEEE, 2020, pp. 542–550.

[14] K. Chen, D. Zhang, L. Yao, B. Guo, Z. Yu, Y. Liu, Deep learning for sensor-based human activity recognition: Overview, challenges, and opportunities, ACM Computing Surveys (CSUR) 54 (4) (2021) 1–40.

[15] M. Alessandrini, G. Biagetti, P. Crippa, L. Falaschetti, C. Turchetti, Recurrent neural network for human activity recognition in embedded systems using ppg and accelerometer data, Electronics 10 (14) (2021) 1715.

[16] G. Bhat, Y. Tuncel, S. An, H. G. Lee, U. Y. Ogras, An ultra-low energy human activity recognition accelerator for wearable health applications, ACM Transactions on Embedded Computing Systems (TECS) 18 (5s) (2019) 1–22.

[17] Y. L. Coelho, F. d. A. S. d. Santos, A. Frizera-Neto, T. F. Bastos-Filho, A lightweight framework for human activity recognition on wearable devices, IEEE Sensors Journal 21 (21) (2021) 24471–24481. doi:10.1109/JSEN.2021.3113908.

[18] P. Mayer, M. Magno, L. Benini, Energy-positive activity recognition - from kinetic energy harvesting to smart self-sustainable wearable devices, IEEE Transactions on Biomedical Circuits and Systems 15 (5) (2021) 926–937. doi:10.1109/TBCAS.2021.3115178.

[19] K. Wang, J. He, L. Zhang, Sequential weakly labeled multiactivity localization and recognition on wearable sensors using recurrent attention networks, IEEE Transactions on Human-Machine Systems 51 (4) (2021) 355–364.

[20] K. Wang, J. He, L. Zhang, Attention-based convolutional neural network for weakly labeled human activities' recognition with wearable sensors, IEEE Sensors Journal 19 (17) (2019) 7598–7604.

[21] F. Samie, L. Bauer, J. Henkel, Hierarchical classification for constrained iot devices: A case study on human activity recognition, IEEE Internet of Things Journal 7 (9) (2020) 8287–8295.

[22] F. Daghero, D. J. Pagliari, M. Poncino, Two-stage human activity recognition on microcontrollers with decision trees and cnns, in: 2022 17th Conference on Ph. D Research in Microelectronics and Electronics (PRIME), IEEE, 2022, pp. 173–176.

[23] M. Odema, N. Rashid, M. A. Al Faruque, Eexnas: Early-exit neural architecture search solutions for low-power wearable devices, in: 2021

IEEE/ACM International Symposium on Low Power Electronics and Design (ISLPED), IEEE, 2021, pp. 1–6.

[24] N. Rashid, B. U. Demirel, M. A. Al Faruque, Ahar: Adaptive cnn for energy-efficient human activity recognition in low-power edge devices, IEEE Internet of Things Journal (2022).

[25] D. N. Jha, Z. Chen, S. Liu, M. Wu, J. Zhang, G. Morgan, R. Ranjan, X. Li, A hybrid accuracy-and energy-aware human activity recognition model in iot environment, IEEE Transactions on Sustainable Computing (2022).

[26] N. Ellis, S. Cittolin, L. Mapelli, Signal processing, triggering and data acquisition, New Technologies for Supercolliders (1991) 361–385.

[27] L. Huang, M. Garofalakis, J. Hellerstein, A. Joseph, N. Taft, Toward sophisticated detection with distributed triggers, in: Proceedings of the 2006 SIGCOMM workshop on Mining network data, 2006, pp. 311–316.

[28] K. Rana, R. Singh, K. Sayann, Correlation based novel technique for real time oscilloscope triggering for complex waveforms, Measurement 43 (3) (2010) 299–311.

[29] J. Wirsich, A. P. Bagshaw, M. Guye, L. Lemieux, C.-G. Bénar, Experimental design and data analysis strategies, in: EEG-fMRI: Physiological Basis, Technique, and Applications, Springer, 2023, pp. 267–322.

[30] A. Bulling, U. Blanke, B. Schiele, A tutorial on human activity recognition using body-worn inertial sensors, ACM Computing Surveys (CSUR) 46 (3) (2014) 1–33.

[31] J.-L. Reyes-Ortiz, L. Oneto, A. Samà, X. Parra, D. Anguita, Transition-aware human activity recognition using smartphones, Neurocomputing 171 (2016) 754–767.

[32] J. Wang, Y. Chen, S. Hao, X. Peng, L. Hu, Deep learning for sensor-based activity recognition: A survey, Pattern recognition letters 119 (2019) 3–11.

[33] S. Mekruksavanich, A. Jitpattanakul, LSTM networks using smartphone data for sensor-based human activity recognition in smart homes, Sensors 21 (5) (2021) 1636.

[34] F. J. Ordóñez, D. Roggen, Deep convolutional and LSTM recurrent neural networks for multimodal wearable activity recognition, Sensors 16 (1) (2016) 115.

[35] H. Ma, W. Li, X. Zhang, S. Gao, S. Lu, Attnsense: Multi-level attention mechanism for multimodal human activity recognition., IJCAI (2019) 3109–3115.

[36] O. Banos, J.-M. Galvez, M. Damas, H. Pomares, I. Rojas, Window size impact in human activity recognition, Sensors 14 (4) (2014) 6474–6499.

[37] J. Cheng, O. Amft, P. Lukowicz, Active capacitive sensing: Exploring a new wearable sensing modality for activity recognition, International conference on pervasive computing (2010) 319–336.

[38] M. M. Hassan, M. Z. Uddin, A. Mohamed, A. Almogren, A robust human activity recognition system using smartphone sensors and deep learning, Future Generation Computer Systems 81 (2018) 307–313.

[39] C. Hou, A study on IMU-based human activity recognition using deep learning and traditional machine learning, 2020 5th International Conference on Computer and Communication Systems (ICCCS) (2020) 225–234.

[40] N. Ketkar, Introduction to keras, in: Deep learning with Python, Springer, 2017, pp. 97–111.

[41] L. Li, K. Jamieson, G. DeSalvo, A. Rostamizadeh, A. Talwalkar, Hyperband: A novel bandit-based approach to hyperparameter optimization, The Journal of Machine Learning Research 18 (1) (2017) 6765–6816.

[42] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, et al., Tensorflow: A system for large-scale machine learning, in: 12th {USENIX} symposium on operating systems design and implementation ({OSDI} 16), 2016, pp. 265–283.

[43] G. B. Team, Tensorflow lite for microcontrollers (2022).
URL `https://www.tensorflow.org/lite/microcontrollers`

[44] G. Weiss, WISDM Smartphone and Smartwatch Activity and Biometrics Dataset , UCI Machine Learning Repository, DOI: https://doi.org/10.24432/C5HK59 (2019).

[45] A. Alevizaki, N. Trigoni, watchHAR: A Smartwatch IMU dataset for Activities of Daily Living (Sep. 2022). doi:10.5281/zenodo.7092553.
URL https://doi.org/10.5281/zenodo.7092553

[46] R. . Schwarz, Ngmo2 datasheet (2020).
URL https://www.rohde-schwarz.com/it/brochure-scheda-tecnica/ngmo2/

[47] National.Instruments, Pc-6251 datasheet (2020).
URL http://www.ni.com/pdf/manuals/375213c.pdf

[48] N. Instruments, Installation guide bnc-2120 (2020).
URL http://www.ni.com/pdf/manuals/372123d.pdf

[49] M. Sokolova, G. Lapalme, A systematic analysis of performance measures for classification tasks, Information processing & management 45 (4) (2009) 427–437.

Declaration of interests

☒ The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

☐ The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: