UNICA

UNIVERSITÀ
DEGLI STUDI
DI CAGLIARI

**Ph.D. DEGREE IN**

MATHEMATICS AND COMPUTER SCIENCE

Cycle XXXV

# TITLE OF THE Ph.D. THESIS

*Machine learning techniques for sensor-based household activity recognition and forecasting*

Scientific Disciplinary Sector(s)

INF/01


| | |
|---|---|
| Ph.D. Student: | Marco Manolo Manca |
| Supervisor | Prof. Daniele Riboni |
| Co-Supervisor | Prof. Ernesto Bonomi |


Final exam. Academic Year 2021/2022
Thesis defence: April 2023 Session

# Abstract

Thanks to the recent development of cheap and unobtrusive smart-home sensors, ambient assisted living tools promise to offer innovative solutions to support the users in carrying out their everyday activities in a smoother and more sustainable way. To be effective, these solutions need to constantly monitor and forecast the activities of daily living carried out by the inhabitants. The Machine Learning field has seen significant advancements in the development of new techniques, especially regarding deep learning algorithms. Such techniques can be successfully applied to household activity signal data to benefit the user in several applications.

This thesis therefore aims to produce a contribution that artificial intelligence can make in the field of activity recognition and energy consumption. The effective recognition of common actions or the use of high-consumption appliances would lead to user profiling, thus enabling the optimisation of energy consumption in favour of the user himself or the energy community in general. Avoiding wasting electricity and optimising its consumption is one of the main objectives of the community. This work is therefore intended as a forerunner for future studies that will allow, through the results in this thesis, the creation of increasingly intelligent systems capable of making the best use of the user's resources for everyday life actions.

Namely, this thesis focuses on signals from sensors installed in a house: data from position sensors, door sensors, smartphones or smart meters, and investigates the use of advanced machine learning algorithms to recognize and forecast inhabitant activities, including the use of appliances and the power consumption. The thesis is structured into four main chapters, each of which represents a contribution regarding Machine Learning or Deep Learning techniques for addressing challenges related to the aforementioned data from different sources.

The first contribution highlights the importance of exploiting dimensionality reduction techniques that can simplify a Machine Learning model and increase its

efficiency by identifying and retaining only the most informative and predictive features for activity recognition. In more detail, it is presented an extensive experimental study involving several feature selection algorithms and multiple Human Activity Recognition benchmarks containing mobile sensor data.

In the second contribution, we propose a machine learning approach to forecast future energy consumption considering not only past consumption data, but also context data such as inhabitants' actions and activities, use of household appliances, interaction with furniture and doors, and environmental data. We performed an experimental evaluation with real-world data acquired in an instrumented environment from a large user group.

Finally, the last two contributions address the Non-Intrusive-Load-Monitoring problem. In one case, the aim is to identify the operating state (on/off) and the precise energy consumption of individual electrical loads, considering only the aggregate consumption of these loads as input. We use a Deep Learning method to disaggregate the low-frequency energy signal generated directly by the new generation smart meters being deployed in Italy, without the need for additional specific hardware.

In the other case, driven by the need to build intelligent non-intrusive algorithms for disaggregating electrical signals, the work aims to recognize which appliance is activated by analyzing energy measurements and classifying appliances through Machine Learning techniques. Namely, we present a new way of approaching the problem by unifying Single Label (single active appliance recognition) and Multi Label (multiple active appliance recognition) learning paradigms. This combined approach, supplemented with an event detector, which suggests the instants of activation, would allow the development of an end-to-end NILM approach. The proposed approach exploits feature extraction techniques that allow the detection of both activated/deactivated appliances and all active appliances, given the aggregated current and voltage signals.

All the proposed techniques have been experimented with real-world data, achieving promising results. Hence, the contributions presented in this thesis pave the way to innovative applications and future research directions. Furthermore, all the algorithms presented below and the analyses performed can, with appropriate additions and modifications, provide continuous monitoring of activities or energy consumption, useful for the user to monitor expenditure and for the energy supplier

to optimise incoming load.

# Contents

# Chapter 1

# Introduction

The recent development of new and effective machine learning techniques has enabled several novel applications to support the user in everyday life. One of the fields in which these techniques see an interesting and useful application is the one of smart homes. In fact, machine learning is an efficient solution to many problems of ambient assisted living, providing powerful tools for activity recognition and monitoring of usage of household appliances. Ambient intelligence-enabled smart homes would provide an effective support to everyday life, and in some cases may reduce the environmental impact. Sensors installed in homes are an excellent source of data useful for training new algorithms capable of predicting users' activities, routines, and habits. Smart meters, position sensors, door sensors and smartphones provide signals and data that, when processed by artificial intelligence algorithms, may increase the ability to monitor and predict the inhabitants' behaviour.

However, existing sensor-based activity recognition systems are prone to several issues regarding limited accuracy of predictions, complexity of the trained model, and overfitting. This thesis provides innovative contributions to the above mentioned limitations of sensor-based activity recognition and forecasting, proposing novel techniques for feature extraction and supervised reasoning with sensor data. Furthermore, all the proposed algorithms have been tested on real-world datasets, which highlights the applicability of our research and the direction to follow in future developments of the field.

The main topics covered in this work are four. Chapter 3 concerns *Human Activity Recognition* and provides an analysis of the various feature selection methods that can be used to tackle this problem. Sensor-based *Human Activity Recognition*

(HAR) is a growing research field that deals with automatically identifying the activities a user is performing based on the analysis of data collected from a variety of sensors [1]. HAR systems have several important applications in different areas including ambient assisted living [2], activity monitoring for health assessment [3], assistance for child and elderly care [4] as well as assistance for people with cognitive disorders [5]. In particular, sensors fitted in mobile devices (accelerometers, gyroscopes, etc.) have now reached widespread adoption and allow to envisage intelligent monitoring systems that can seamlessly track daily activities, in a non-intrusive way, and help users to make better decisions about their future actions [6]. However, such an automatic decision support capability is still limited, despite the increasing capacity of processing data from smart devices. Furthermore, the exploitation of HAR systems in real-world scenarios not only demands high recognition accuracy but also poses multiple challenges in terms of power consumption and robustness with respect to different context conditions [7, 8], soliciting further research efforts in this field.

Sensors installed in a house can be used to collect data in order to predict future energy consumption. The power forecasting field addresses this problem; in Chapter 4, our work offers a solution to this problem by training a machine learning algorithm with data from various types of sensors. In recent years, there has been increasing interest in forecasting electricity consumption in residential buildings [9, 10, 11]. Indeed, homes are a major source of energy consumption, especially in urban areas. Forecasting energy consumption in buildings may thus help improving the smart grid, reducing costs and emissions. Energy forecasting is also useful to support human activities in ambient intelligence systems [12].

An alternative to power forecasting, in terms of user benefit, can be provided by electricity monitoring. Non-Intrusive Load Monitoring (NILM), or load disaggregation, aims to identify the operating state (on/off) and precise energy consumption of individual electrical loads, considering only the aggregate consumption of these loads as input. The NILM technique was introduced by Hart's pioneering work in the mid-1980s [13]. In fact, he was the first to use measured active and reactive power time series to estimate the on/off status of individual devices. In Chapter 5 our work aims to present a deep learning based algorithm for disaggregating electrical power on filtered signals from a household smart meter. This method could be useful for monitoring electricity consumption for the consumer himself, thereby optimising the

electrical load in the grid and the household's consumption. Research on this topic has had a strong push in the last decade, due to new requirements in power grid management systems and thanks to the developments in machine learning and deep learning techniques.

One of the main sub-tasks of Non-Intrusive Load Monitoring (NILM) is appliance recognition. Its aim is to recognize which appliance is activated by analysing energy measurements and classifying appliances through machine learning techniques. This task can be approached in two ways: the first one is to recognise a single appliance by using its activation signals, the second one is to recognise which appliances are on and which are off by analysing the aggregated electrical signals. Several works address these issues [14, 15, 16]; however, research works concentrate on addressing them independently of each other. In Chapter 6 we try to solve the two problems simultaneously by combining two deep neural networks and the use of some feature extraction techniques allow us to understand which appliances are being activated and which are already active by knowing the aggregated current and voltage values.

All the proposed techniques have been experimented with real-world sensor data. The results that we obtained are promising, and claim for further investigation of the proposed methods. The thesis concludes in Chapter 7 with concluding remarks and promising directions for future work.

# Chapter 2

# State of the art

The purpose of this chapter is to report the main works presented in the literature on the topics covered in this thesis. The literature described in this chapter is intended to give a general overview of the topics studied in the remainder of this document; however, a more specific state of the art will be provided in each subsequent chapter.

## 2.1 Feature selection on human activity recognition

More and more often in the field of research, one has to deal with a large amount of data and, for this reason, new feature selection techniques are always being researched in order to better manage the available data. Even in the field of HAR, several works have investigated effective feature selection methods. Chetty G. et al. [17] in their work use an information theory-based feature classification algorithm to recognise human activity from the inertial sensors of smartphones. Gupta P. et al. [18] similarly use feature selection techniques called Relief-F and sequential forward floating search (SFFS) to increase the performance of their activity recognition algorithm, which uses data from a body-worn wireless accelerometer. Some works, such as that of Ahmed N. et al. [19], combine a Support Vector Machine classifier with a hybrid feature selection system, which includes a filter and wrapper method, with the aim of recognising human activity from smartphone data. Amjad et al. [20] also presents a two-level hierarchical method for human activity recognition using a set of wearable sensors. In their work, they investigate feature extraction

methods based on atomic scores, testing on signals from smartphones, smartwatches and smart glasses.

In the same context, there are also several comparative studies that provide an idea of which feature selection techniques are most effective to use in this field. For example, Chong J. et al. [21] analyse the effects of different feature subsets on the predictive performance of the most popular Machine Learning algorithms; the main objective is to identify which combination of feature subset and predictive algorithm performs best for the prediction of activity classes from raw acceleration data. Amezzane et al. [22] in their work also present a comparison of feature selection methods with the aim of reducing the training and classification time, of the algorithms that can be used in the field of HAR, as much as possible. Also Suto J. et al. [23] present a paper in which common filter feature selection techniques compared with a Bayesian wrapper feature selection method in order to demonstrate that the wrapper technique outperforms filter algoritmhs in the case of HAR problems.

In other cases, machine learning techniques are used for feature selection. Chen et. al [24] in fact adopt Random Forest to select important features in the classification of human activity and other financial assets. Bashar S. K. et al. [25] on the other hand use neighbourhood component analysis to select a subset of important features from the available parameters in the time and frequency domain; then the classification of human activities is carried out using a dense neural network. Game theory can also be used in the field of feature selection, in fact Guha R. et al. [26] developed a method, called Cooperative Genetic Algorithm, with the aim of selecting the most important features for visual human action recognition, in order to improve accuracy and lower the activity recognition time.

## 2.2 Power forecasting

In recent decades, the energy problem has certainly attracted research interest and many important works have been provided about power forecasting. In this context, Völker et al. [27] provide a comprehensive review of the state of the art of smart meter data analysis applications and try to provide an overview of the technological foundations and the impact on future developments.

Many works have investigated algorithms for predicting energy consumption. Dong et al. [28] propose a short-term consumption forecasting algorithm for res-

idents using reactive power. Haq et al. [29] use a hybrid algorithm that exploits machine learning techniques to predict electricity consumption of household appliances and peak demand.

Several researches in this field also use some deep learning techniques to predict energy consumption. An example is the work of Shi et al. [30] in which a deep neural network based on pooling is trained to cluster a group of customers' load profiles into a pool of inputs. S. Ai [31] also exploits a neural network ensamble to obtain the electricity demand forecasts of multiple households in three scenarios, optimising potential network configuration set, forecasting single household power demand, and refilling missing data.

## 2.3 Non-intrusive-load-monitoring

Recently, the literature has seen an incremental interest in the NILM problem. In fact, several works attempt to tackle this problem using increasingly sophisticated techniques. Deep learning is one of the most widely used methodologies to solve the disaggregation problem, exploiting neural networks of various types.

Harell A. et al. [32] use a casual 1-D convolutional neural network on low-frequency data in order to disaggregate the aggregate power of 20 sub-meters. They also investigate the benefit of using all four energy components, current, active power, reactive power, and apparent power, available in NILM datasets. Similarly, in the work of Kelly J. et al. [33] the disaggregation of the electrical power is performed by three different neural networks. The first one is a recurrent neural network called 'Long Short-Term Memory' (LSTM), the second one is a denoising autoencoder and the last one is a regressor that predicts the activation and deactivation time of each appliance and thus provides the average power consumption per single appliance. The combination of sliding windows and two Recurrent neural networks is used in the work of Krystalakos O. et al. [34], their proposal disaggregates the electrical power signal of 5 different appliances by knowing the aggregated low-frequency signal. The implemented technique is tested on a real-world dataset.

In [35], Gomes E. et al. propose the application of the pinball quantile loss function to guide deep neural networks, such as Convolutional Neural Networks and Recurrent Neural Networks, to tackle the power disaggregation problem.

Another approach used in the field of NILM is the Bayesian one, for instance

Culiere F. et al. [36] use a Bayesian model of temperature-dependent electricity consumption. This model allows the heating component to be disaggregated from the electrical load curve in an unsupervised manner.

## 2.4 Appliance recognition

As already described in the Introduction to this thesis, appliance recognition is an important branch of NILM, which is why much of the literature in this field uses techniques very similar to those described in the previous section. The use of machine learning techniques is certainly the route taken most often by researchers in this field. Both de Paiva Penha D. et al.[37] and Rehmani M. et al. [38] use such techniques for the recognition of residential equipment. The former train a convolutional neural network and test the proposed algorithm on a real low-frequency dataset, the latter evaluate the effectiveness of the most commonly used deep learning techniques on a set of real datasets.

In a different way but in the same context, Nalmpantis C. et al. [39] focus their work on the dimensionality reduction by using a technique called Signal2Vec; in this way they analyse multi-label NILM systems and propose a framework capable of offering cost-effective solutions.

The above-mentioned works share the intention to recognise the activation status of all devices in a home, but this is not the only approach with which to tackle the problem of appliance recognition. In fact, there is a different approach, which consists to recognise a single appliance by knowing the activation signal. Faustine A. et al. [40, 41] in two papers studied important feature extraction techniques, namely Weighted Recurrence Graphs, to train a Convolutional Neural Network capable of recognising which household appliance was switched on. Their methods are validated on real high-frequency datasets containing the current and voltage signals of each household appliance.

# Chapter 3

# Exploiting feature selection in human activity recognition: methodological insights and empirical results using mobile sensor data

Human Activity Recognition (HAR) using mobile sensor data has gained increasing attention over the last few years, with a fast-growing number of reported applications. The central role of machine learning in this field has been discussed by a vast amount of research works, with several strategies proposed for processing raw data, extracting suitable features, and inducing predictive models capable of recognizing multiple types of daily activities. Since many HAR systems are implemented in resource-constrained mobile devices, the efficiency of the induced models is a crucial aspect to consider. This chapter highlights the importance of exploiting dimensionality reduction techniques that can simplify the model and increase efficiency by identifying and retaining only the most informative and predictive features for activity recognition. More in detail, a large experimental study is presented that

This chapter is published as M. M. Manca, B. Pes and D. Riboni, "Exploiting Feature Selection in Human Activity Recognition: Methodological Insights and Empirical Results Using Mobile Sensor Data", in IEEE Access, vol. 10, pp. 64043-64058, 2022, doi: 10.1109/ACCESS.2022.3183228.

encompasses different feature selection algorithms as well as multiple HAR benchmarks containing mobile sensor data. Such a comparative evaluation relies on a methodological framework that is meant to assess not only the extent to which each selection method is effective in identifying the most predictive features but also the overall stability of the selection process, i.e., its robustness to changes in the input data. Although often neglected, in fact, the stability of the selected feature sets is important for a wider exploitability of the induced models. Our experimental results give an interesting insight into which selection algorithms may be most suited in the HAR domain, complementing and significantly extending the studies currently available in this field.

## 3.1 Introduction

In recent years, several machine learning approaches have been explored for the automatic classification of daily living activities based on mobile sensing [42, 17, 43, 44, 45]. The overall process involves different steps, including the cleansing of raw data to remove noise and artifacts, and the extraction of high-level features that can be useful to discriminate among the considered activities [46]. A common methodology for feature extraction relies on segmenting the sensor data, e.g., the tri-axial acceleration signals, into time windows that are subsequently mapped, through proper functions, into a set of meaningful features. Different types of mapping approaches have been explored based on the application scenario [47], resulting in time-domain, frequency-domain, or other types of features that can be finally fed to a classifier to induce the HAR model. Automatic feature extraction based on deep learning methods has also been recently investigated [7]. Both the features' definition and the choice of the classifier may significantly affect the performance of the recognition system, as witnessed by a vast amount of literature in the field [3, 6, 47].

The computational efficiency of the HAR system is another important aspect to consider when dealing with mobile devices that may have limitations in terms of processing capability as well as energy consumption. From this point of view, it may be convenient to contain the dimensionality of the feature vectors used at the classification stage, as discussed in some recent studies [18, 48, 19]. The feature extraction process can indeed result in a large number of features, especially in a

multi-sensor system where the feature sets generated from the single sensors can be fused to create feature vectors of high dimensionality [49]. In such a scenario, it can be useful to apply automatic techniques to identify and select the most important features for prediction, in order to simplify the activity recognition models, decrease overfitting, and reduce computations. Indeed, it has been observed that not all the extracted features have the same importance in the classification of physical activities [46, 50]. Some features may be redundant, weakly relevant, or even irrelevant/noisy for a specific task, suggesting that feature selection techniques could be effectively employed to reduce the data dimensionality and improve the HAR system's efficiency without compromising the final prediction accuracy.

However, not much research has so far explored the potential of feature selection in this field. Most emphasis has been given to investigating which classification methods and strategies may be best suited for inducing the HAR models [3, 47, 51, 52], while few studies have compared the impact of different feature selection algorithms on such models [21], gaining insight into which heuristics may be most effective in selecting reduced subsets of discriminative features. To make a contribution in this direction, we present here an extensive comparative study that encompasses several selection approaches and investigates their behavior on typical HAR datasets extracted from mobile devices, like smartphones and smartwatches, where recognition performance and computational efficiency need to be jointly optimized.

More in detail, extending our previous research in this field [53], this work provides a two-fold contribution. First, a general methodological framework is presented that allows evaluating the effectiveness of the feature selection process along with two directions: *i)* the capacity of the algorithm of identifying the most important features for activity prediction, and *ii)* the stability of the selected feature subsets, i.e., their robustness to perturbations in the input data. This way we can better understand the suitability of a given selection method for the considered application scenario. Indeed, identifying feature subsets that are both highly predictive and stable is important for a wider exploitability of the induced models, as highlighted in recent literature [54, 55].

Leveraging such a framework, we carried out an experimental evaluation for different levels of dimensionality reduction, i.e., for different percentages of selected features, in order to find an optimal trade-off between the number of features used

for prediction and the resulting classification performance. Specifically, our study includes selection methods that are representative of different heuristics: *univariate* approaches, which assess every single feature independently of the others; *multivariate* approaches, which capture the inter-dependencies among the features; *filter* approaches, which carry out the selection process without interacting with the learning algorithm; *embedded* approaches, which exploit the features' weights derived by a proper classifier to assess the relevance of the features. Such a comprehensive evaluation has been performed in conjunction with learning algorithms that have proven highly effective in this domain, such as *Support Vector Machines* and *Random Forest*.

The experimental study has been conducted on five HAR datasets extracted from mobile sensor data [56, 57, 58, 59, 60], both using a single sensor type (accelerometer) or different types of sensors (accelerometer, gyroscope, magnetometer). Further, the considered benchmarks present different levels of dimensionality, ranging from about 150 features to over a thousand features, which has allowed us to explore the behavior of the considered selection algorithms across different feature spaces.

Overall, the results of our experiments show that the feature selection process can reduce the original dimensionality to a great extent without any degradation in the final recognition performance, which confirms the importance of introducing an automatic dimensionality reduction step into any mobile sensing processing pipeline. By jointly evaluating the predictive performance and the selection stability, we also obtained some interesting insights into which methods may be best suited in the considered domain. To the best of our knowledge, this is the first work that evaluates several feature selection approaches in this field based on different real-world benchmarks. Moreover, in this work we investigate the stability of selection methods, which is neglected by most of the existing studies.

The remainder of this work is structured as follows. Section 3.2 gives some background concepts on feature selection and discusses its applications in the HAR field. Section 3.3 describes the adopted methodological framework and the specific selection algorithms chosen for the experimental study. The characteristics of the considered datasets are illustrated in Section 3.4. Section 3.5 presents the experimental analysis and, finally, Section 3.6 further discusses the main findings and gives some concluding remarks as well as directions for future work.

## 3.2 Background concepts and related work

In the last two decades, several research efforts have focused on devising proper methods for handling high dimensional datasets [61]. Feature selection plays an important role in such a context as it can discard irrelevant and redundant information, as well as noisy factors, with significant benefits in terms of computational efficiency, model interpretability and data understanding [62]. As summarized below, a wide variety of feature selection methods can be found in the literature, with some promising applications also in the HAR field [17, 18, 48, 19, 21, 63, 64, 65, 22, 24, 25, 66, 67, 23].

### 3.2.1 Background on feature selection

In the context of supervised learning tasks, like those considered in this chapter, the available selection techniques can be broadly distinguished into three categories [62, 68]:

- *Filters* methods, that conduct the selection process as a pre-processing step, without interacting with the learning algorithm used at the model induction stage, thus leading to a classifier-independent selection outcome. Only the intrinsic characteristics of the training data are taken into account to estimate the relevance of the features, i.e., the extent to which they can be useful in separating the different classes. This can involve the *individual evaluation* of every single feature, based on its correlation with the class attribute, or the *evaluation of subsets* of features, within which the reciprocal correlation among the features is also considered to minimize redundancy (but at an increased computational cost).

- *Wrapper* methods, that search for the feature subset that can optimize the predictive performance of a given classifier. In this case, the learning algorithm itself is employed as an evaluation function to assess each candidate subset of features. Such an approach involves inducing a model from each subset and measuring the resulting performance through a proper validation protocol (e.g., a cross-validation procedure internal to the training set). The computational cost of the selection process is hence dependent on the classifier's intrinsic efficiency, as well as on the search strategy used to build the

candidate subsets (e.g., an evolutionary search or a greedy stepwise search), with an overall burden generally higher than the one of filter methods.

- *Embedded* methods, that leverage the intrinsic capacity of some classification algorithms (e.g., *Perceptron* classifiers or linear *Support Vector Machines* classifiers) to assign weights to the features, without requiring a systematic search through different candidate subsets, as in the case of wrappers. In terms of computational cost, this approach often provides a reasonable trade-off between filters and wrappers, with results that have proven quite satisfactory in multiple scenarios.

Several studies have investigated the potential and the drawbacks of the different selection methods proposed so far [69, 70]. Due to their reduced computational requirements, filter and embedded methods have found wider adoption in high-dimensional problems, with a variety of algorithms available that support both *univariate* and *multivariate* selection processes, as better discussed in section 3.3. Hybrid and ensemble techniques, that properly integrate different selection methods, have also been studied with promising results in recent years [62, 71, 72]. There is, however, no selection approach that outperforms the others in all situations and choosing the most appropriate method for a given task remains often difficult [73].

### 3.2.2 Feature selection in the HAR field

Feature selection methods falling in the filter category have been generally preferred in the HAR field. For example, [63] and [64] have investigated the use of *MRMR* and *CFS* filters [61], which are designed to search for subsets of features that are highly correlated with the target class but not correlated with each other. Such a subset-oriented evaluation tries to reduce redundancy and can be an effective and viable solution when the original dimensionality is not too high. A more efficient ranking-based filter is exploited in [17], where every single feature is weighted according to an information theoretical criterion, known as *Information Gain*, with an overall ordering of features based on the resulting weights. Such an approach allows discarding the features that are less useful in discriminating the target class and has proven well suited even in the presence of very high dimensionalities. Ranking-based filters have also been applied in [65], where the *Chi-Squared*, *Fisher score*, and *ReliefF* methods are employed to weight the features and arrange them in decreasing

order of relevance. A comparison between ranking-based and subset-oriented filters is presented in [48], which emphasizes that the filter-selected, classifier-independent, feature subsets have potentially broad exploitability in smartphone-based HAR.

Fewer applications can be found in this field for the embedded methods and the wrapper methods [22, 24, 25]. Specifically, given the higher computational cost of wrappers, they have been mainly applied after preliminarily reducing the data dimensionality by means of a filter, as in [19] and [66]. Some direct comparisons between filter and wrapper approaches are presented in [18, 67, 23]; in such studies, however, the dimensionality of the original feature space is relatively low, which can make acceptable the higher computation time required by wrappers.

Despite an increasing amount of research pointing out the benefits of feature selection in HAR applications, there is a lack of comparative studies that extensively evaluate the strengths and weaknesses of the different selection approaches in this field. At the time of writing, the largest experimental comparison is presented in [21], where ten different selection algorithms (seven filters, two wrappers and one embedded method) are evaluated on a single sensor dataset involving 206 attributes (both time-domain and frequency-domain features); interestingly, the feature subsets selected using efficient filter methods are found to outperform those produced by wrappers that, even though potentially able to produce superior results, are more prone to the problem of overfitting.

Finally, to the best of our knowledge, only the impact of feature selection on the performance of HAR models has been considered so far, without investigating the stability of the selection process, i.e., its sensitivity to changes in the input data, which may critically affect the robustness of the induced models. Taking such an aspect into account, this work presents a wide comparative analysis that complements the studies available in this field, encompassing several selection methods and several benchmarks, as detailed in the following sections.

## 3.3 Methodological framework

Our methodological framework is meant to be general enough to be implemented with different selection methods as well as different learning algorithms. Specifically, it relies on a ranking-based selection approach that allows controlling the level of dimensionality reduction at a fine-grained level, in order to find the optimal trade-off

between model efficiency and classification performance. This is indeed a primary concern in the application domain here considered. Further, as anticipated above, our methodology involves evaluating both the predictive power and the stability of the selected feature subsets, which is important to understand the extent to which these subsets can be truly relevant for the task at hand, regardless of the specific composition of the training data. All the steps of the adopted methodology are outlined in what follows, along with a description of the specific methods and settings chosen for the comparative analysis.

### 3.3.1 Ranking-based feature selection and performance evaluation

As a general framework for a wide comparison, we chose a ranking-based selection approach [62] that is flexible enough to encompass the use of different selection algorithms, including *filter* methods, that assign weights to the features based on their degree of correlation with the class (i.e., the activity to be predicted), and *embedded* methods, that rely on the features' weights derived by a proper learning algorithm. The assigned weights, regardless of how they are calculated, can be used to obtain a *ranked list* in which the $N$ features of the data at hand ($D$) appear in decreasing order of relevance, i.e., from the most important (rank 1) to the least important (rank $N$), as schematized in Figure 3.1. Such a list can then be cut at a suitable threshold point ($n$) to select a subset of highly relevant features, i.e., the $n$ top-ranked features.

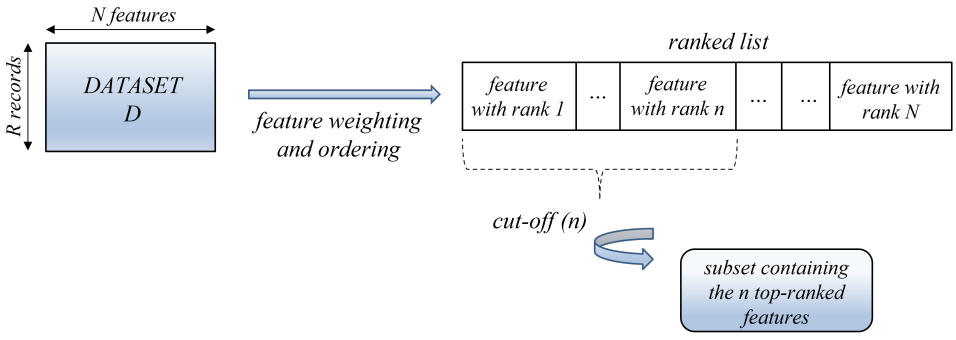


Figure 3.1: The adopted ranking-based selection approach

Considering only these features, an activity recognition model can be induced from $D$ by training any suitable classifier. The adopted ranking-based selection

framework is indeed not tied to a specific classification method, which makes it potentially suitable for a variety of implementation scenarios, as discussed in section 3.5 (where different implementation settings are explored). The performance of the resulting models is evaluated on a separate set of test records that are structured to contain only the features previously selected from $D$ (to avoid any selection bias, in fact, the test records must not be used in the feature selection process).

The best level of dimensionality reduction, i.e., the optimal value of the cut-off threshold in Figure 3.1, may vary depending on the specific task at hand. The effect of modifying such a threshold is explored experimentally in our study, which encompasses different values of $n$ and evaluates their impact on the final recognition performance. This approach allows discarding the unnecessary features in a cost-effective way, especially when the data dimensionality makes impractical the direct adoption of wrapper-based search strategies (but such strategies, if needed, could be subsequently applied for refining the outcome of the selection process and optimizing it for the chosen classifier).

### 3.3.2 Stability evaluation

Recent literature has highlighted the importance of investigating the stability of feature selection with respect to variations in the input data [54, 74]. For a stable method, we expect to obtain (almost) the same outcome when the original set of training instances is somewhat perturbed (e.g., randomly removing a given percentage of records).

Evaluating stability essentially involves two aspects [75]: *(i)* a procedure to create multiple sample sets from the available data, and *(ii)* a consistency index to quantify the sensitivity of the selection process to sample variation. More in detail, given a dataset $D$ with $R$ instances and $N$ features, a number $K$ of reduced datasets $D_i$ (i=1,2,...,$K$) are drawn, each containing a fraction $f$ of the original instances. The chosen selection algorithm is then applied to each $D_i$, obtaining an output $O_i$ (i=1,2,...,$K$) that may depend on $D_i$'s specific composition. A proper similarity measure is finally used to assess the pairwise similarity between the outputs $O_i$: the more their average similarity, the more stable the selection method.

In our framework, each output $O_i$ takes the form of a feature subset $S_i$ containing $n$ of the original features (selected according to the approach explained previously), for a total of $K$ subsets of the same size. To evaluate how similar these subsets are to

each other, we rely on a consistency index known as *Kuncheva measure* [76], which has proved to be suitable in the context of high dimensional problems. Specifically, given a pair of subsets $S_i$ and $S_j$, their similarity is measured as follows:

$$sim_{ij} = \frac{|S_i \cap S_j| - n^2/N}{n - n^2/N} \tag{3.1}$$

where $|S_i \cap S_j|$ is the number of features that are common to $S_i$ and $S_j$. The similarity $sim_{ij}$ essentially measures the degree of overlap between the two subsets, with a proper correction reflecting the probability that a feature is included in both subsets simply by chance (such a probability increases as the subset size $n$ approaches the original dimensionality $N$).

The resulting similarity values are then averaged across all pair-wise comparisons to assess the overall degree of consistency among the $K$ subsets:

$$sim_{avg} = \frac{2}{K(K-1)} \sum_{i=1}^{K-1} \sum_{j=i+1}^{K} sim_{ij} \tag{3.2}$$

This average similarity can be assumed as a measure of the stability level of the selection process. Since $sim_{ij}$ and $sim_{avg}$ may vary in dependence on the size $n$ of the selected subsets, our experimental study investigates the stability trend for feature subsets of increasing size, as shown in Section 3.5.

### 3.3.3 Methods and settings

For implementing the methodological framework presented above, we considered some popular selection algorithms that are representative of different feature weighting paradigms. In particular, we employed five *univariate methods*, that weigh every single feature independently of the others, and five *multivariate* methods, that are able to capture the interdependencies among the features.

Specifically, among the univariate techniques, we chose:

- *Chi Squared* ($\chi^2$), that leverages the well-known chi-squared statistic to evaluate how relevant a feature is with respect to the class [69]. Specifically, once

a feature has been discretized into $I$ intervals, its $\chi^2$ value is obtained as:

$$\chi^2 = \sum_{i=1}^{I}\sum_{j=1}^{C} \frac{(A_{ij} - \frac{R_i \cdot B_j}{R})^2}{\frac{R_i \cdot B_j}{R}} \tag{3.3}$$

where $R$ is the total number of instances, $C$ the number of classes, $R_i$ the number of instances in the $ith$ interval, $B_j$ the number of instances in the $jth$ class, and $A_{ij}$ the number of instances in the $ith$ interval and $jth$ class.

- *Information Gain* ($IG$), that measures the extent to which the class entropy decreases when the value of a given feature is known: the greater the decrease in entropy, the more discriminative the feature [77]. Namely, by denoting as $H$ the entropy function, we can derive the $IG$ value for a feature $X$ as:

$$IG(X) = H(Y) - H(Y|X) \tag{3.4}$$

where $H(Y)$ is the entropy of the class $Y$ before observing $X$, while $H(Y|X)$ is the conditional entropy of $Y$ given $X$ [78].

- *Symmetrical Uncertainty* ($SU$) and *Gain Ratio* ($GR$), that in turn rely on the $IG$ measure but include suitable correction factors that try to compensate for the $IG$'s bias toward features with more values [79]. Specifically, after computing the $IG$ value for a feature $X$, the corresponding $SU$ and $GR$ values are obtained as follows:

$$SU(X) = \frac{2 \cdot IG(X)}{H(X) + H(Y)} \tag{3.5}$$

$$GR(X) = \frac{IG(X)}{H(X)} \tag{3.6}$$

where $H(X)$ and $H(Y)$ denote the entropy of, respectively, the feature $X$ and the class $Y$.

- *OneR* ($OR$), that weights each feature based on the accuracy of a simple classification rule built on that feature, according to the approach originally proposed in [80]. More in detail, for each of the available features, the algorithm creates one rule by determining the most frequent class for each feature's value (a rule is simply a set of attribute values bound to their majority class).

The prediction accuracy of each rule is then computed, and the features are ranked according to the quality of the corresponding rules.

Among the multivariate techniques, we considered two *Relief*-based selectors [81] and three *SVM*-based selectors [82]. More in detail, we chose:

- *ReliefF* (*RF*), that evaluates the strength of the features according to their ability to discriminate between data instances that are near to each other (nearest neighbors) in the feature space. Basically, in the original two-class formulation, a sample instance $R_i$ is extracted from the training set, and its features' values are compared to the corresponding values of the instance's nearest *hit* $H$ (neighbor from the same class) and *miss* $M$ (neighbor from the opposite class). A weight $W$ is then iteratively computed for each feature $X$, starting from an initial value $W(X) = 0$:

$$W(X) := W(X) - \frac{diff(X, R_i, H)}{r} + \frac{diff(X, R_i, M)}{r} \tag{3.7}$$

  where $r$ is the number of randomly drawn instances and $diff$ is a function that computes the difference between the value of $X$ for $R_i$ and $H$ as well as for $R_i$ and $M$. The underlying assumption is that a "good" feature should have the same value for data points of the same class and different values for data points of different classes. Such a binary formulation can be easily extended to deal with multi-class problems [81]. In turn, *ReliefF-weighted* (*RFW*) adopts a similar strategy but weighting the neighbors by their distance.

- *SVM-AW*, that relies on a linear *SVM* classifier, which has an embedded capability of assigning a weight to each feature based on how it contributes to the hyperplane decision function induced by the classifier. This function can indeed be written as follows:

$$f(\boldsymbol{X}) = \boldsymbol{W} \cdot \boldsymbol{X} + b \tag{3.8}$$

  where $\boldsymbol{X}$ is the $N$-dimensional vector of input features, $\boldsymbol{W}$ is a weight vector, and $b$ is a bias constant. The weight $W_j$ assigned to the $jth$ feature can be interpreted as a measure of the strength of the feature; specifically, the *SVM-AW* algorithm considers the absolute value of this weight (*AW*) [82].

- *SVM-RFE*, that, in turn, exploits a linear *SVM* but adopts a *recursive feature elimination* strategy that iteratively removes the features with the lowest weights and repeats the weighting process on the remaining features. In our study, two versions of this approach are evaluated: *RFE10*, where the percentage $p$ of features removed at each iteration is set to 10%, and *RFE50*, where this percentage is 50% (in the special case where $p = 100\%$, *SVM-RFE* reduces to *SVM-AW* as no iteration occurs).

As regards the computational complexity of the above techniques, it depends on both the number of features ($N$) and the number of instances ($R$). In particular, it can be shown that the number of operations is of the order of $N \cdot R$ for the univariate approaches [83], while the multivariate approaches have a higher computational cost. Indeed, in the worst case, the number of operations is of the order of $N \cdot R^2$ for the *Relief*-based methods while it is of the order of $max(N, R) \cdot R^2$ for *SVM-RFE*. However, efficient implementations exist that optimize the nearest-neighbor calculations involved in *ReliefF* as well as the kernel-matrix calculations involved in *SVM-RFE* [84].

Furthermore, note that some of the above techniques ($\chi^2$, *IG*, *GR*, *SU*, *RF*, and *RFW*), which only rely on the data's intrinsic characteristics, fall in the category of filter methods, while others (*OR*, *SVM-AW*, *RFE10*, and *RFE50*) leverage the features' weights derived by a suitable classifier and can be thus categorized as embedded methods. Irrespective of the specific algorithm used to derive the features' weights, the final selection is carried out according to the approach shown in Figure 3.1, i.e., by retaining a number $n$ of top-ranked features; such a number is varied in our experiments encompassing different percentages of selected features, from 5% to 90%.

As learning algorithms to induce the activity recognition models, we chose, after a series of preliminary experiments, an *SVM* classifier with a polynomial kernel of degree 2 and a *Random Forest* classifier, which proved to be a suitable option for the considered benchmarks (described in Section 3.4). For both classifiers, as well as for the different selection methods, we leveraged the implementations provided by the *WEKA* machine learning library [85].

More in detail, we trained the *SVM* classifier using the well-known *Sequential Minimal Optimization* (*SMO*) algorithm [86]. For the *Random Forest* classifier, we relied on 100 unpruned trees, each built using $\log_2(n)+1$ random features at the split-

ting stage, according to commonly adopted settings [87]. The *WEKA ChiSquaredAttributeEval*, *InfoGainAttributeEval*, *SymmetricalUncertAttributeEval*, *GainRatioAttributeEval*, and *OneRAttributeEval* were used to implement the univariate methods $\chi^2$, *IG*, *SU*, *GR*, and *OR*, respectively. For the *Relief*-based methods, we employed the *ReliefFAttributeEval* function, with and without instance weighting (for *RFW* and *RF* respectively). Finally, for the *SVM*-based selection methods, i.e., *SVM-AW* and *SVM-RFE*, we exploited the *SVMAttributeEval* function, properly setting the percentage of features to be removed at each iteration. Each of these feature weighting functions was used in conjunction with the *Ranker* search method that allows selecting a specified number of top-ranked features.

## 3.4 Datasets

For our comparative study, we used five datasets meant for mobile human activity recognition. Specifically, one of these datasets contains data acquired from a fitness watch and a mobile phone, three of them contain smartphone sensor data, and the last one contains body-worn sensor data. In each dataset, data instances are labeled with the corresponding activities, which are primarily sport/fitness activities and activities of daily living. The high-level characteristics of these experimental benchmarks are summarized in Table 3.1.

The ***COSAR*** dataset [56, 44] was collected in experiments concerning the recognition of 10 different activities: *brushing teeth*, *climbing up and down*, *riding bycicle*, *standing still*, *jogging and strolling*, *walking downstairs and upstairs*, *writing on blackboard*. These activities were carried out by 6 volunteers wearing two accelerometers: the first one located inside the left pocket and the second one on the right wrist. In addition, a GPS receiver in the left pocket tracked the person's location. Overall, the dataset consists of 5 hours of activity data, sampled at a frequency of 16Hz, and each activity instance has a time extension of 1 second, for a total of 18000 instances (divided into 13500 training records and 4500 test records, as reported in Table 3.1).

The second dataset (***HAR***) [57] was collected from 30 volunteers wearing a smartphone on the waist, as described in [88]. Inertial data were acquired from the smartphone's 3-axial accelerometer and 3-axial gyroscope at 50 Hz. Each subject carried out six activities: *walking*, *walking upstairs and downstairs*, *sitting*, *stand-*

Table 3.1: Datasets used in the experimental study.

| Dataset | Number of records | Number of features | Number of classes | Sensors |
|---|---|---|---|---|
| **COSAR**∗ | 18000 (13500 training + 4500 test) | 148 | 10 | 2 accelerometers |
| **HAR** ∗∗ | 10299 (7352 training + 2947 test) | 561 | 6 | 1 accelerometer, 1 gyroscope |
| **HAR_ALL** ∗∗∗ | 5744 (4252 training + 1492 test) | 561 | 6 | 1 accelerometer, 1 gyroscope |
| **HAPT** ∗∗∗∗ | 10929 (7767 training + 3162 test) | 561 | 12 | 1 accelerometer, 1 gyroscope |
| **DSA** ∗∗∗∗∗ | 9118 (6838 training + 2280 test) | 1170 | 19 | 5 accelerometers, 5 gyroscopes, 5 magnetometers |

∗ *Everywarelab Activity Recognition Dataset*

∗∗ *Human Activity Recognition Using Smartphones Dataset*

∗∗∗ *Smartphone Dataset for Human Activity Recognition in Ambient Assisted Living*

∗∗∗∗ *Smartphone-Based Recognition of Human Activities and Postural Transitions Dataset*

∗∗∗∗∗ *Daily and Sports Activities Dataset*

*ing and laying*. Raw data were filtered to remove noise and sampled using a 50% overlapping sliding window of 2.56 seconds, resulting in a total of 10299 instances (partitioned into 70% training data and 30% test data).

The third dataset (***HAR_ALL***) [58] is an extension of the *HAR* dataset described above; indeed, it was obtained by carrying out similar experiments and contains the same activities, as described in [2]. 30 subjects participated in the experiments, resulting in a collection of 5744 records.

In turn, the ***HAPT*** dataset [59] is an updated version of the *HAR* dataset that contains an extended set of activities, including postural transitions: *walking*, *walking upstairs and downstairs*, *sitting*, *standing and laying*, *stand-to-sit and stand-to-lie*, *sit-to-stand and sit-to-lie*, *lie-to-sit and lie-to-stand*. This benchmark, described in detail in [43], is considered especially challenging due to the highly imbalanced activity distribution (given the lower frequency of postural transitions).

The last dataset we considered is (***DSA***) [60], previously used in various research works including [89] and [90]. It contains 19 daily and sport activities: *sitting*, *standing and lying on back and on right side*, *ascending and descending stairs*, *standing and moving around in an elevator*, *walking in a parking lot and on a treadmill (in flat and 15˚ inclined positions)*, *running on a treadmill*, *exercising on a stepper and on a cross trainer*, *cycling on an exercise bike in horizontal and vertical positions*, *rowing*, *jumping and playing basketball*. The activities were carried out by 8 subjects and the

total duration of each activity was 5 minutes per subject. Data were acquired at a sampling rate of 25 Hz, using five body-worn orientation trackers, each containing a 3-axial accelerometer, a 3-axial gyroscope and a 3-axial magnetometer. The sensor signals were divided into 5-second segments, yielding a total of 9120 instances (480 per activity).

As we can see in Table 3.1, the five considered benchmarks have quite different dimensionalities, due to the different numbers of sensors involved as well as to the different set of high-level features computed for each sensor signal (after segmenting it into time windows). Specifically, for each window, feature vectors were computed in the time and frequency domains (*Fast Fourier Transform* was applied to sensor signals to transform data in the frequency domain). Several statistical measures were computed for each window: some of them, e.g., mean, standard deviation, Kurtosis, and min/max values, were extracted for all five datasets, while others have been used only in some of them. Note that we maintained the features' definitions employed in the original studies in which the datasets were published, in order to avoid introducing any bias and make the experiments fully repeatable. See Appendix A for more details on the extracted features.

## 3.5 Experimental analysis

Leveraging the activity recognition benchmarks described in the previous section, we conducted an extensive experimental study aimed at investigating the extent to which the original feature space can be reduced without degrading the final predictive performance. The overall analysis, in terms of both capacity of discriminating the classes and selection stability, has been carried out for different levels of dimensionality reduction, according to the methodological framework detailed in Section 3.3. The main experimental results are summarized in Figures 3.2, 3.3 and 3.4, each containing ten charts that compare the behavior of the considered selection methods.

Specifically, Figure 3.2 and Figure 3.3 show a comparison in terms of *F-score*, which is a performance metric widely employed in activity recognition tasks. Defined as the harmonic mean between the model *sensitivity* (i.e., the fraction of positive instances classified correctly) and the model *precision* (i.e., the fraction of correct predictions among all the instances assigned to the positive class), the F-score takes

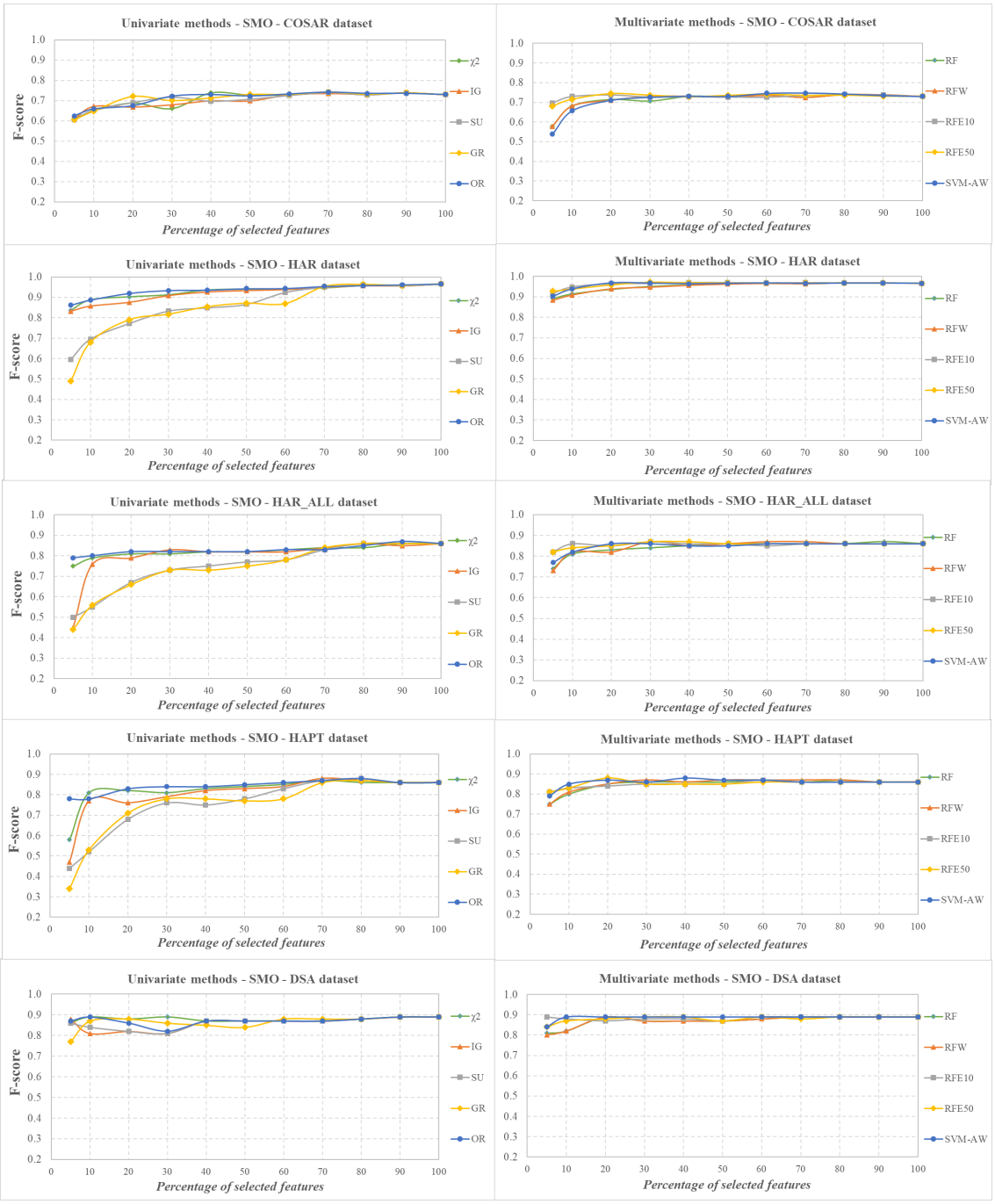


Figure 3.2: F-score performance of the *SMO* classifier, in conjunction with the univariate (on the left) and the multivariate (on the right) selection methods.

both the false positives and the false negatives into account, providing a reliable estimate of the model ability to recognize a given class (considered as positive). By measuring the average F-score across the different classes, we obtained an overall

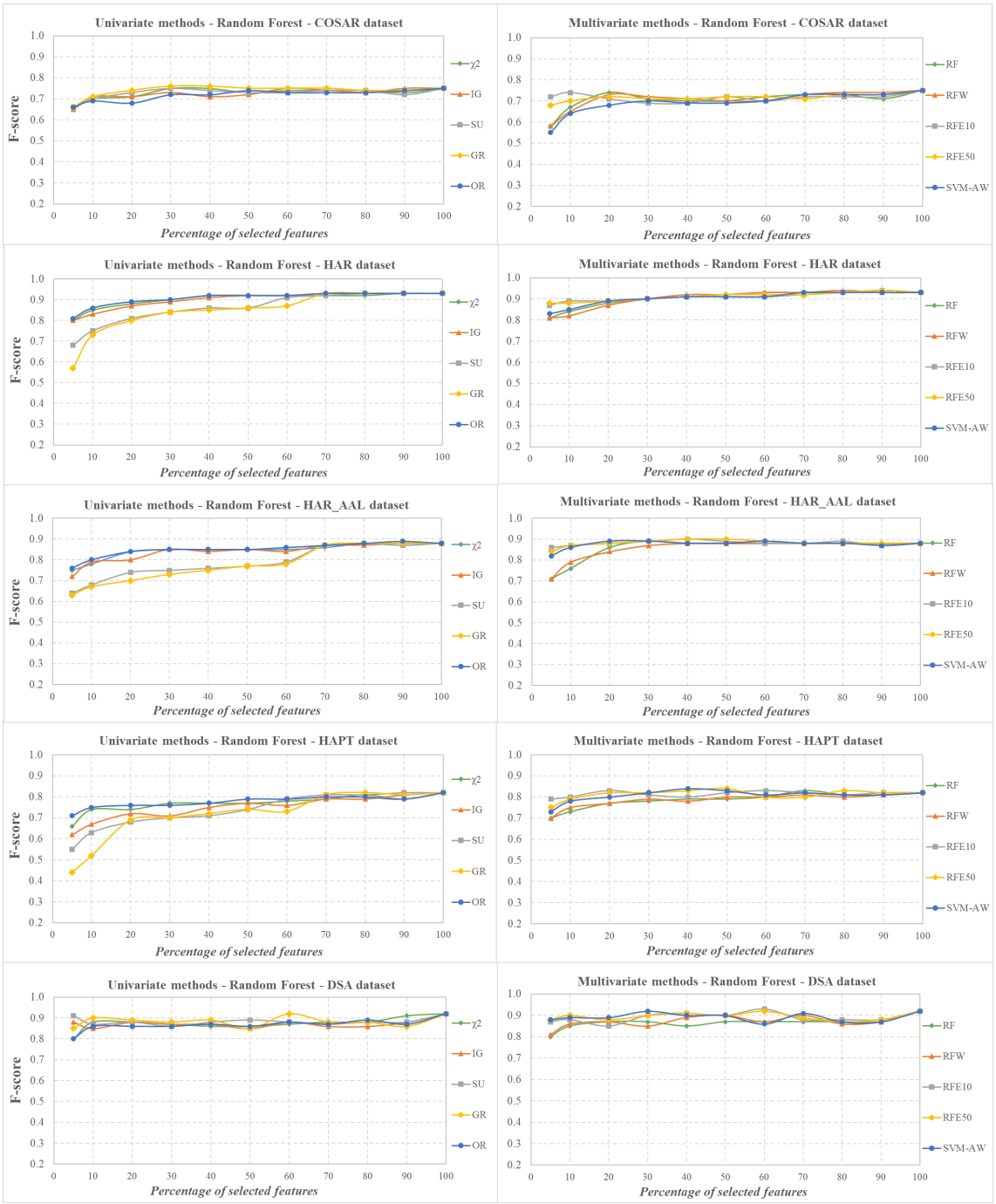


Figure 3.3: F-score performance of the *Random Forest* classifier, in conjunction with the univariate (on the left) and the multivariate (on the right) selection methods.

evaluation of the recognition performance of the induced models. For model induction, as anticipated in Section 3.3.3, we employed both the *SMO* (Figure 3.2) and

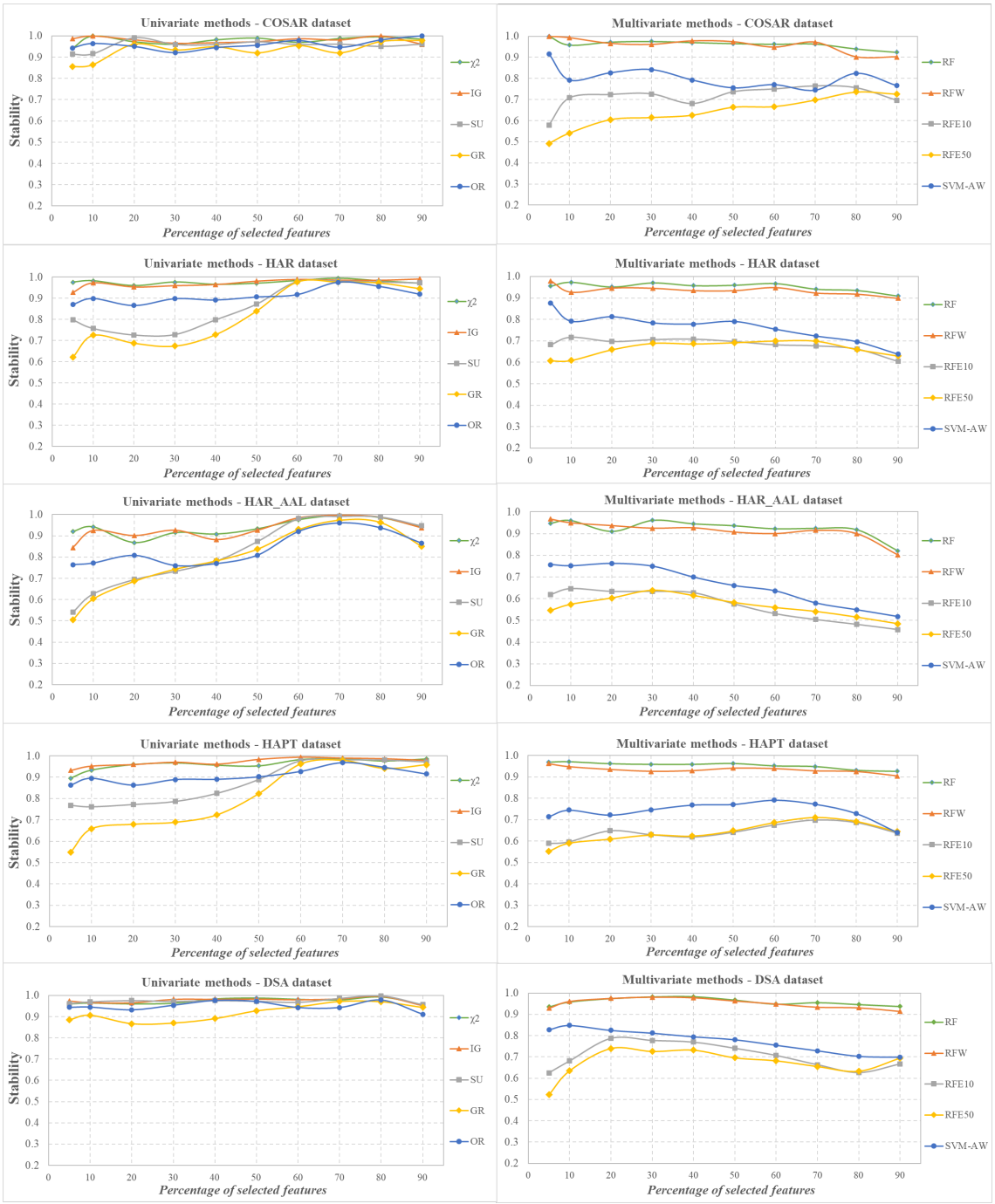


Figure 3.4: Stability trend for the univariate (on the left) and the multivariate (on the right) selection methods.

*Random Forest* (Figure 3.3) classifiers, in conjunction with ten different selection methods.

More in detail, for both the univariate ($\chi^2$, *IG*, *SU*, *GR*, *OR*) and the multivariate (*RF*, *RFW*, *RFE10*, *RFE50*, *SVM-AW*) selection techniques, Figures 3.2 and 3.3 show the F-score trend for different percentages of selected features (the abscissa 100 corresponds to the model induced on the whole dataset, without any dimensionality reduction). In absolute terms, the *COSAR* dataset is the one where both *SMO* and *Random Forest* achieve the lowest recognition performance, due to the intrinsic difficulty of discriminating multiple activities using features extracted only from accelerometer signals. In the other four datasets, where we have a multi-sensor scenario (as detailed in Table 3.1), the average F-score is generally better, sometimes above 0.9, with quite similar trends for the two classifiers.

But the most interesting observation, for the purpose of our study, is that no significant degradation in performance is observed when the original dimensionality is reduced, regardless of the specific characteristics of the dataset at hand. In particular, 10-20% (or even less) of the original features may be sufficient, for some selection methods, to obtain recognition performances comparable to those achieved using the whole dataset. This reveals the appropriateness of introducing a suitable feature selection step into any activity recognition protocol in order to simplify the final models and make them more efficient.

When comparing the different selection techniques, the multivariate approaches, which can capture the inter-dependencies among the features, turn out to be overall more effective in terms of F-score. In particular, among the *SVM*-based methods, *RFE10* and *RFE50* sometimes have slightly superior performances but at a higher computational cost than the simpler *SVM-AW*, whose behavior is still satisfactory. As regards the *Relief*-based multivariate approaches (*RF* and *RFW*), quite good performance can be obtained with at most 20% of the features. Among the univariate methods, on the other hand, *SU* and *GR* overall exhibit the worst behavior, with an unsatisfactory performance for some datasets, especially when small feature subsets are selected. For the other univariate approaches, i.e., $\chi^2$, *IG* and *OR*, feature subsets containing (at most) 20% of the features turn out to be sufficient to obtain quite good F-score values.

Overall, the strong potential of feature selection in this domain is witnessed by both Figures 3.2 and 3.3, despite some small differences between the curves reported for the *SMO* and *Random Forest* classifiers. However, as recognized by recent literature in the feature selection field, e.g., [54, 62, 91, 75], a good selection

method should not only be effective (in terms of final predictive performance) but also as stable as possible to avoid that the selected subsets depend too much on the specific composition of the training data, thus becoming less useful in future applications of the model.

The results of the stability analysis we have performed on the five considered benchmarks (*COSAR*, *HAR*, *HAR_AAL*, *HAPT*, *DSA*) are shown in Figure 3.4, where the stability trend is reported for different percentages of selected features, according to the methodology detailed in sub-section 3.3.2; specifically, such a methodology has been here implemented with the settings $K = 20$ and $f = 0.80$, which have proven suitable in similar studies [54, 62].

A first point to highlight regarding Figure 3.4 is that the differences observed in terms of stability are generally higher than those observed in terms of F-score (Figure 3.2 and Figure 3.3), revealing that some selection methods systematically exhibit a more stable behavior across the different datasets. In particular, $\chi^2$ and *IG* turn out to be the most stable of the univariate approaches, followed by *OR*, while *SU* and *GR* have shown significantly lower stability in some datasets; *GR*, in particular, appears to be the least robust method in the univariate group. Among the multivariate approaches, on the other hand, the *Relief*-based methods (i.e., *RF* and *RFW*) have proven to be very stable, with similar trends for all levels of dimensionality reduction. Conversely, the *SVM*-based methods appear to be less robust, especially *RFE10* and *RFE50*, whose stability is always lower than the one of the simpler *SVM-AW*.

Overall, the univariate $\chi^2$, *IG* and even *OR* show a quite good trade-off between F-score and stability and may therefore be an option to consider when inducing activity recognition models from mobile sensor datasets like those considered in our study. As well, the multivariate *Relief*-based approaches exhibit a good behavior when jointly considering both predictive performance and stability, at least for subsets containing more than 10% of the original features. It is worth mentioning, nevertheless, that the least stable methods could be made significantly more robust when implemented in a bootstrap-based ensemble version [62], thus envisaging margins of improvement for some selection techniques (but at the expense of the computational cost of the feature selection process).

Based on a wide set of experiments, the above results consolidate the findings of our preliminary study in this field [53], showing that feature selection can be

effectively exploited to reduce the dimensionality of mobile sensor datasets, leading to robust and more efficient recognition models.

## 3.6 Discussion

The experimental analysis here presented complements and extends the findings of recent comparative studies conducted on similar activity recognition benchmarks [48, 65, 92, 21], providing stronger evidence of the suitability of the filter selection paradigm in this field.

Specifically, [48] discusses the advantages of leveraging classifier-independent selection approaches that can identify feature subsets conveying useful information for targeted populations and applications, regardless of the chosen classifier and the specific implementation settings. Compared to our work, the dataset considered in their study is relatively low-dimensional, with only seventy-six features, which allows efficiently using subsets-oriented filters (*CFS*, *FCBF*), along with a multivariate ranker (*ReliefF*), while the ranking-based approach is usually preferred in the presence of higher dimensionalities [65, 92]. Both subset-oriented filters (*CFS*, *MRMR*) and ranking-based filters (*Information Gain*, *Gain Ratio*, *Symmetrical Uncertainty*, *ReliefF*), along with wrapper and embedded methods, are also evaluated in [21], using a single sensor dataset involving 206 attributes (both time-domain and frequency-domain features), with empirical evidence of the effectiveness of the simpler univariate rankers in yielding good feature subsets for HAR models.

Through wider experiments on five high-dimensional benchmarks, our study has presented a more in-depth evaluation of the ranking-based selection approach, with both univariate and multivariate techniques, showing that they can be effectively employed in conjunction with different classifiers (see Figures 3.2 and 3.3). Actually, besides filters that inherently perform a classifier-independent selection, our experiments also encompass ranking methods that leverage some learning algorithm to compute the features' weights, as in the case of the *SVM*-based selectors. The final subsets produced by these methods, of the embedded category, are often used like those produced by filters, i.e., as a reduced feature space to train any potentially suitable classifier. However, as shown in Figure 3.4, this kind of ranker has proved to be overall less stable.

Our study has also shown that the adopted ranking-based approach allows to

Table 3.2: *HAPT* dataset (1 accelerometer, 1 gyroscope, 561 features): different levels of dimensionality reduction and corresponding F-score performance.

| ***Number of features ($\chi^2$ feature selection)*** | ***Features by sensor type*** | ***F-score (SMO classifier)*** |
|---|---|---|
| 29 (5%) | accelerometer: 29, gyroscope: 0 | 0.58 |
| 57 (10%) | accelerometer: 57, gyroscope: 0 | 0.82 |
| **85 (15%)** | **accelerometer: 81, gyroscope: 4** | **0.85** |
| 561 (100%) | accelerometer: 348, gyroscope: 213 | 0.86 |

control the level of dimensionality reduction in a fine-grained way, in order to find the optimal trade-off between the number of the selected features and the resulting classification performance. Indeed, as discussed in Section 3.5, the original dimensionality can be reduced to a very great extent, with a final recognition performance similar to that achieved with the full feature set.

To provide concrete examples of such a dimensionality reduction, Tables 3.2 and 3.3 show the *SMO* classifier's F-score (averaged across the different classes, as in Figures 3.2 and 3.3) for the two multi-sensor benchmarks with the highest numbers of features/classes. Specifically, besides the full feature set, three feature subsets with smaller cardinality are considered, as selected by one of the univariate rankers that have shown the best tradeoff between predictive performance and stability, namely $\chi^2$ (which also has a reduced computational cost compared to the multivariate ranking methods). As we can see, in the *HAPT* dataset (Table 3.2), only 15% of the original features are sufficient to obtain a final performance comparable to that achieved using the whole feature set, while 10% of the features turn out to be as predictive as the whole feature set in the *DSA* dataset (Table 3.3). Furthermore, interesting insight can be gained into the extent to which each sensor type contributes to the optimal set of features, with clear evidence of the high discriminative power of the features extracted from the accelerometer signals (that, however, are not sufficient to obtain the highest F-score values). This kind of analysis can leverage different ranking methods, as previously observed in Figures 3.2 and 3.3, and allows the design of fast-response recognition systems where only a reduced set of highly discriminative features need to be computed at operation time.

Overall, our results clearly show how beneficial feature selection can be in sensor-based activity recognition. However, some limitations exist in the current study that will be addressed in further investigations. Indeed, the explored ranking-based selec-

Table 3.3: *DSA* dataset (5 accelerometers, 5 magnetometers, 5 gyroscopes, 1170 features): different levels of dimensionality reduction and corresponding F-score performance.

| ***Number of features ($\chi^2$ feature selection)*** | ***Features by sensor type*** | ***F-score (SMO classifier)*** |
|---|---|---|
| 30 (2.5%) | accelerometer: 25, magnetometer: 5, gyroscope: 0 | 0.82 |
| 59 (5%) | accelerometer: 44, magnetometer: 15, gyroscope: 0 | 0.86 |
| **117 (10%)** | **accelerometer: 71, magnetometer: 42, gyroscope:** 4 | **0.89** |
| 1170 (100%) | accelerometer: 390, magnetometer: 390, gyroscope: 390 | 0.89 |

tion approach, although effective and generally more efficient than subset-oriented selection methods, may be sub-optimal in the presence of some degree of redundancy among the features. Hence, the potential impact of feature redundancy in this field should be better analyzed, for example by exploring hybrid selection strategies that first reduce the data dimensionality through a simple and efficient ranker and then further refine the resulting subset through a more sophisticated search strategy that can remove highly correlated features [61]. But this would increase the computational cost of the selection process, requiring careful cost-benefit analysis. Although some recent works have applied a hybrid selection approach in the HAR field [66, 19, 93], there is a lack of comparative studies that investigate the effects of different hybrid strategies in terms of final recognition performance as well as selection stability and computational efficiency.

# Chapter 4

# Towards context-aware power forecasting in smart-homes

Forecasting future power consumption in residential buildings is important to optimize the power grid, to assist inhabitants in everyday activities, and to save energy. Several machine learning methods have been proposed to predict future electricity consumption in smart homes based on the history of past consumption data acquired from smart meters. However, the increasing availability of smart home sensors can provide insights about the routines and activities of inhabitants, that may be exploited to provide more accurate predictions. In this chapter, we propose a machine learning approach to forecast future energy consumption considering not only past consumption data, but also context data such as inhabitants' actions and activities, use of household appliances, interaction with furniture and doors, and environmental data. We performed an experimental evaluation with real-world data acquired in an instrumented environment from a large set of users. The results of a comparison with two baseline methods show that our approach is promising.

This chapter is published as E. Cuncu, M. M. Manca, B. Pes, D. Riboni, "Towards Context-aware Power Forecasting in Smart-homes", in Procedia Computer Science, vol. 198, pp. 243-248, 2022, doi: https://doi.org/10.1016/j.procs.2021.12.235.

## 4.1 Introduction

Several machine learning models have been proposed in the literature to forecast electricity consumption in smart homes. A review of the main artificial intelligence techniques applied to the problem of electrical forecasting in buildings is proposed in [9]. Ahmad et al. describe in detail the performance of Artificial Neural Network and Support Vector Machines, comparing them with classical models such as ARIMA. Similar work has been done by Camara et al. [10], who use two approaches to forecast residential energy consumption. Specifically, they compare predictive performances of Seasonal ARIMA model with Artificial Neural Network, based on historical energy consumption. Mocanu et al. [94] compare three machine learning techniques for time series prediction of energy consumption. Namely, they compare the Conditional Restricted Boltzman Machine (CRBM) with both Artificial Neural Networks and Hidden Markov Models. Their work focuses on CRBM for energy forecasting, using load profiles on measured data.

Other researchers applied deep learning techniques to forecast energy consumption. In their work, Dey et al. [11] provide a comparison between classical time-series forecasting algorithms and a Group Method Data Handling (GMDH) neural network. In that work, the objective was twofold: forecasting energy consumption, and gaining insights about the behaviour and lifestyles of users. Bhatt et al. [95] trained a Convolutional Recurrent Neural Network to find an energy consumption prediction model under different climatic conditions.

Most existing energy forecasting methods rely only on the history of power consumption data, sometimes extended with external data such as temporal information and weather forecasts [96]. A few studies investigated the use of heterogeneous sensor data to improve energy forecasting in smart buildings [97]. Ziekov et al. use wireless sensors, called ZeeBee, to record the electricity consumption in each socket in the smart home. They then apply the collected data to their forecasting methods. One example is the work of Barbato et al. [98], in which a system for predicting the usage states of household appliances is proposed. Dobbe et al. [99] used Bayesian estimation for identifying the optimal number of sensors for the energy forecasting task. Namely, they provide simulation results considering two types of sensors: magnitude sensors and PMUs. Their employment leads to the collection of the voltage magnitude and angle values. In [100], Truong et al. proposed an algorithm to

predict users daily habits and the interdependency of used devices employing the Gibbs sampling procedure. For example, the probability of using a single household appliance during an entire day is used to predict future consumption.

The limitations of the sensors described so far may be related to the fact that they all involve previous electricity consumption. In our work, the sensors used are independent of any energy consumption.

We believe that the increasing availability of sensors in smart homes could provide a valuable data source for improving the forecast of power consumption. In particular, energy consumption is strongly influenced by human activities and other contextual conditions. Hence, heterogeneous context data, including the observation and forecast of human activities, may increase the precision of future energy prediction systems. As a first contribution in this direction, we present a multilayer energy forecasting system that includes context data and activities in the forecasting process. Our system acquires raw sensor data from the smart home infrastructure. It adopts complex reasoning algorithms to recognize actions, activities, and tasks from raw data, and a feature extraction algorithm to build feature vectors using a sliding window. A collaborative learning approach is used to anonymously share context-aware energy data. Those data are used for training machine learning algorithms, including classifiers and regressors, to solve different energy prediction tasks. We implemented a prototype of our system, and performed experiments with a large set of real-world data. The experimental results show that our method clearly improves two baseline algorithms.

The rest of the chapter is structured as follows. Section 4.2 illustrates the architecture of our system. Section 4.3 explains the methods for feature extraction and power load forecasting. Section 4.4 presents our experimental evaluation and the achieved results. Section 4.5 concludes the work.

## 4.2 System overview

Figure 4.1 illustrates an overview of our system. The smart home is instrumented with various sensors to collect data about the inhabitant activities and the environmental conditions. Sensors include position sensors to detect movement of people in the home, and other sensors to detect the interaction with instruments, doors, and items.

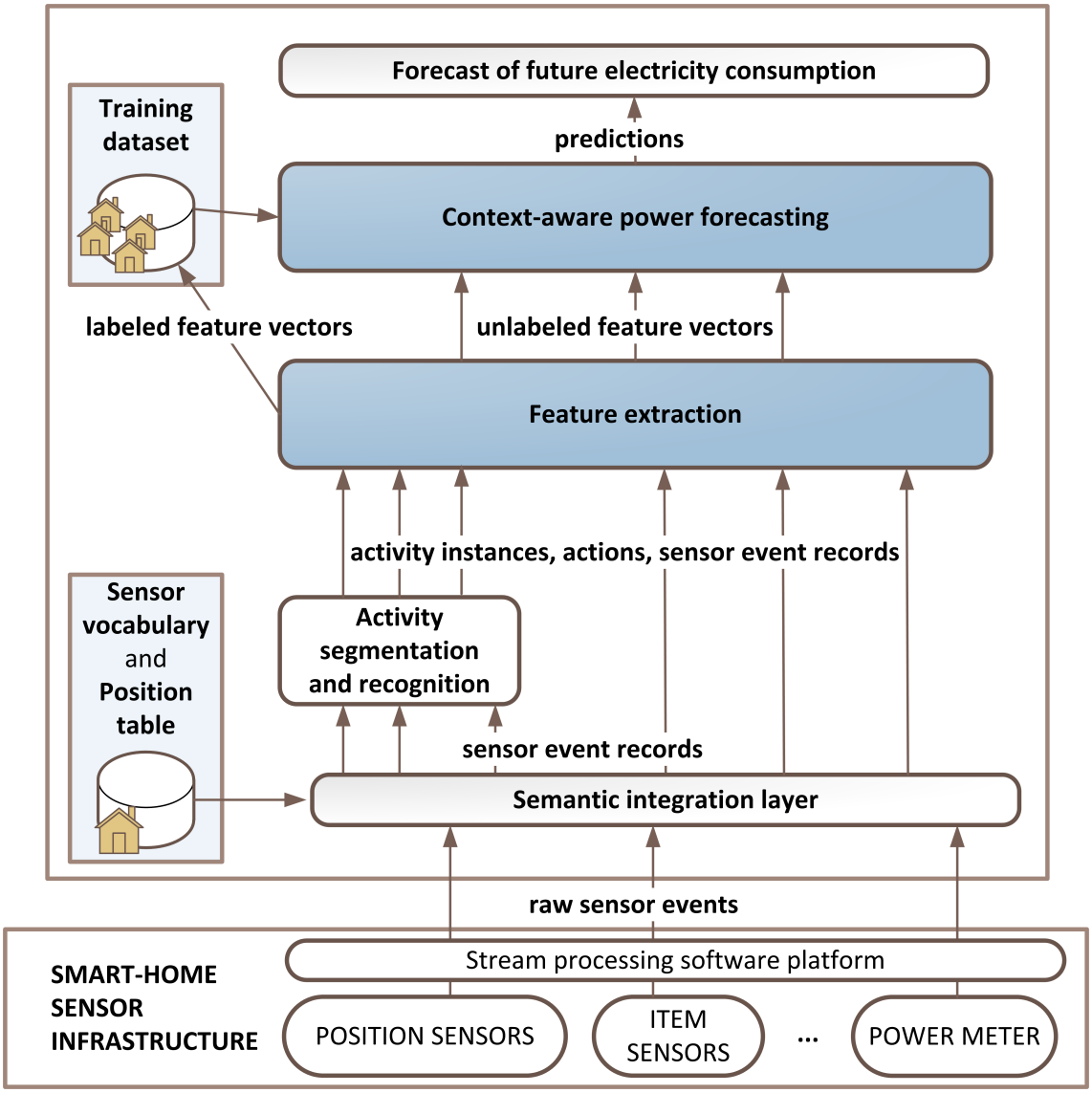


Figure 4.1: System overview.

The infrastructure also includes one power meter that periodically detects instantaneous electricity consumption at the apartment level.

All raw data produced by those sensors are communicated to a 'semantic integration' software layer, which is in charge of assigning a semantics to the data according to a sensor vocabulary. That layer relies on a position table, storing the relative position of each sensor in the home, to determine the position of inhabitants based on fired sensors. Raw sensor data preprocessed by the semantic integration layer are called 'sensor event records'.

A module for activity segmentation and recognition processes the stream of sensor event records for recognizing the activities that are occurring in the home, as well as the actions that compose those activities. That module includes algorithms to segment the activities and identify their start and end time. Indeed, several research efforts have been spent in the last two decades to devise algorithms for activity recognition and segmentation based on sensor data. Different effective solutions to this problem have been proposed, which adopt data-driven [101, 102], knowledge-driven [103, 104], or hybrid methods [105, 106]. Since the goal of this work is power forecasting, in this section we assume the existence of an effective module for action/activity segmentation and recognition, but we do not make any assumption about the actual implementation of that module.

Recognized activities, actions, and sensor event records, including power meter readings, are sent to the module for 'feature extraction', presented in Section 4.3.1. The latter is in charge of extracting statistical features from the data using a fixed-size sliding window. A feature vector is computed for each window.

Feature vectors are communicated to the 'context-aware power forecasting' module, presented in Section 4.3.2. That module is in charge of providing a forecast of future electricity consumption in the smart home. To this aim, the module uses a supervised machine learning algorithm trained on an anonymous dataset acquired in different smart homes.

## 4.3 Methods

In this section, we present our methods for feature extraction and power load forecasting in smart homes.

### 4.3.1 Feature extraction

For each sliding window of $x$ minutes, the module for feature extraction computes the following features:

- Time: the time of the day at the beginning of the window, computed as the number of minutes passed after midnight;
- Temperature: the average temperature registered by a given environmental sensor;

- Activity: for each recognized activity, the number of minutes of that activity execution;
- Presence: for each presence sensor, the number of activations during the window, which is the number of times the sensor has recorded an activity;
- Door: for each door sensor, the number of its activations;
- Action and Task: for each recognized action or task, the number of its executions;
- Trend: the trend of power consumption, computed through linear curve fitting;
- Previous consumption: average electricity consumption during the previous time window;
- Current consumption: average electricity consumption during the current time window.

In addition, the module computes a further value, named 'Future consumption'; i.e., the average power consumption in the following time window. Of course, that value can be computed only with a delay. Feature vectors with future consumption information are communicated to the Training dataset database in anonymous form to enlarge the training set.

### 4.3.2 Power forecasting

The power forecasting module adopts a classical machine learning approach to predict future electricity consumption. Indeed, it uses a dataset of feature vectors, computed as explained in Section 4.3.1 and labeled with the 'Future consumption' value, to train a machine learning algorithm. At run-time, the module uses the trained model to predict future consumption considering unlabeled feature vectors received from the 'feature extraction' module.

The module supports the following kinds of prediction tasks.

- Exact power consumption: in this task, the goal is to predict the average electricity consumption during the next time window. For this task, the module uses a regression machine learning algorithm. More precisely, the module uses

the Random Forest technique, an ensemble learning method that combines many decision trees via bagging [107]. It was decided to use 100 trees and 100 predictors for each split, which is the best performing configuration for this model.

- Power change: in this task, the goal is to predict whether the average electricity consumption during the next time window will be larger or smaller with respect to the current one. Being a binary classification task, in this case the module uses a Logistic Regression classifier, namely a logistic regression model with ridge estimator [108].

All experiments were performed using WEKA software.

The prediction of energy consumption in both regression and classification cases would be valuable information in a real world scenario. In fact, it would make it possible to optimise the load of electricity fed into the grid, thus preventing waste and pollution.

## 4.4 Experimental evaluation

In this section, we report our experimental evaluation and the achieved results.

### 4.4.1 Dataset and experimental setup

We performed our experiments using a dataset acquired and labeled by researchers of the Center for Advanced Studies in Adaptive Systems[1] (CASAS) of Washington State University. The data were acquired from smart home sensors while different categories of people, including persons with cognitive diseases, were carrying out everyday tasks [109]. Since we do not target persons with cognitive disabilities, in our experiments we used only the data acquired from cognitively healthy subjects. Totally, we acquired data from 305 subjects, each performing activities in the smart home for approximately 4 hours. The CASAS smart home is a two-story apartment. The first floor of the apartment consist of a kitchen, living room and a dining area, while bathroom and two bedrooms are located at the second floor.

[1] http://casas.wsu.edu/

The smart home is equipped with several passive infrared (PIR) motion sensors mounted on the ceiling to track the user's position. It is instrumented with several sensors, including item sensors to detecting the usage of selected kitchen tools, door sensors, burner sensors, hot and cold water sensors, temperature sensors. It also includes a whole-apartment electricity usage meter. In total, the smart-home includes 52 motion sensors, 18 door sensors, 10 item sensors, and one smart meter. The dataset is labeled with the activities, actions and tasks performed by the user.

We have experimented our method using different sizes of the sliding window, ranging from 5 minutes to 150 minutes. For building the feature vectors, we used the raw sensor data and the ground truth annotations of actions, activities and tasks. We have one feature for time, 5 features for average temperature values in different rooms of the home, 52 features for presence sensors, 18 features for door sensors, 16 features for activities, 13 features for tasks, 12 features for actions, one feature for trend, one feature for previous consumption, and one feature for current consumption.

We ordered the feature vectors according to the start time of the corresponding sliding window; we used the first half of them as the training set, and the second half as the test set.

The Figure 4.2 represents the average energy consumption for each user, the horizontal line in blue indicates the average value of all averages. It seems evident from the graph that the consumption values are distributed rather evenly.

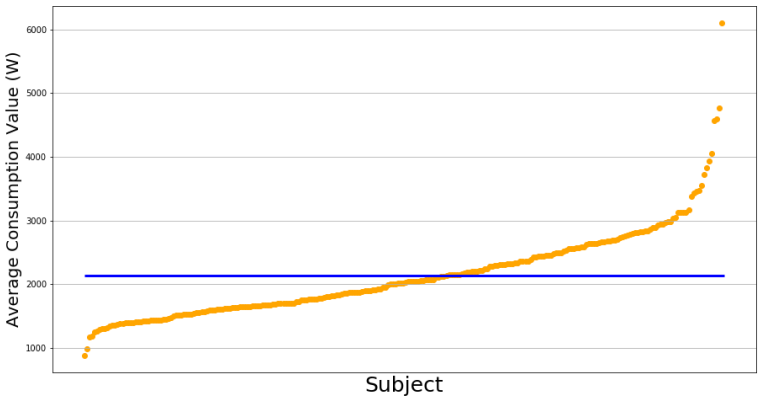


Figure 4.2: Average Consumption Values per subject.

### 4.4.2 Results

We have compared our method with two baselines. The first one, named 'stationary', assumes that the average power consumption in the next time window is exactly the same of the current one. The second baseline, named 'trend prediction', assumes that power consumption in the home follows a linear trend. Hence, with this baseline, we forecast the average consumption in the next time window based on the linear interpolation of the power consumption readings of the current window. For the sake of this work, we do not apply feature selection techniques.

The experimental results are shown in Figures 4.3 and 4.4. The results of the 'Exact power consumption' task, shown in Figure 4.3, are expressed in terms of Pearson correlation coefficient [110].With this evaluation metric it was possible to see how well the algorithm estimated the energy consumption trend, not just the exact consumption value. The results show that our method clearly outperforms the baselines with time windows of 100 minutes or less. With longer time windows, the 'stationary' baseline achieves results close to the one of our methods.

The results of the 'Power change' task, shown in Figure 4.4, are expressed in terms of accuracy; i.e., the percentage of correctly classified labels. Accuracy is an adequate metric in our case, since classes ('increase' vs 'decrease') are balanced. For this experiment, we did not use the 'stationary' baseline, since it could not predict neither an increase or a decrease of power consumption. Results show that our method clearly outperforms the 'trend prediction' baseline.

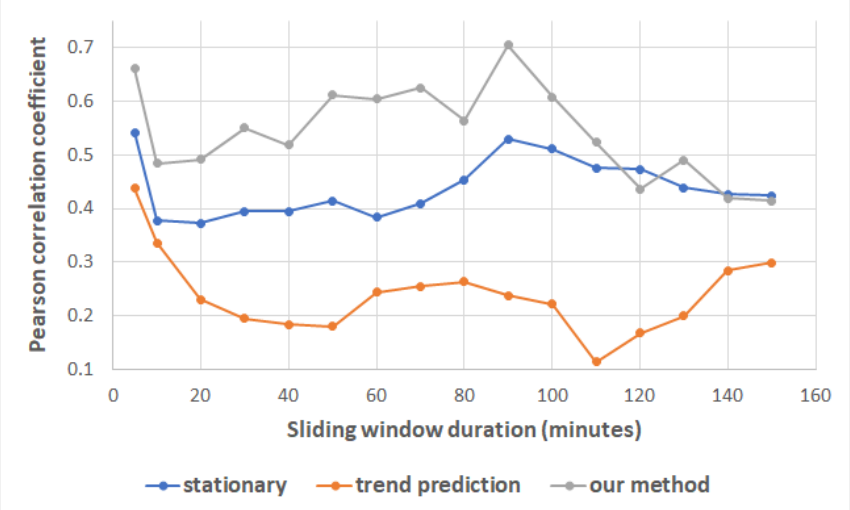


Figure 4.3: Experimental results: exact power consumption

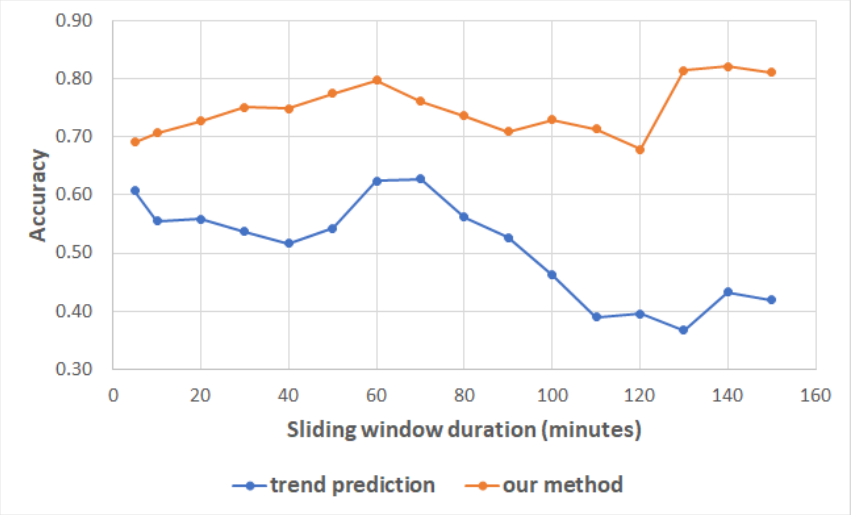


Figure 4.4: Experimental results: power change

## 4.5 Discussion

An algorithm capable of predicting future electricity consumption in a smart home was presented in this chapter. The strategy of sliding windows as a feature extraction method led to promising results. Two types of approaches were evaluated; in one case, the focus was to estimate the actual consumption, tackling a regression problem; in the other case, the problem was studied from a classification point of view, determining a prediction of an increase or decrease in consumption.

To benchmark our algorithm, we used a public dataset to train and test the models. The calculated evaluation metrics allow us to confirm the validity of our strategy.

# Chapter 5

# Deep learning based non-intrusive load monitoring with low resolution data from smart meters

A detailed knowledge of the energy consumption and activation status of the electrical appliances in a house is beneficial for both the user and the energy supplier, improving energy awareness and allowing the implementation of consumption management policies through demand response techniques. Monitoring the consumption of individual appliances is certainly expensive and difficult to implement technically on a large scale, so non-intrusive monitoring techniques have been developed that allow the consumption of appliances to be derived from the sole measurement of the aggregate consumption of a house. However, these methodologies often require additional hardware to be installed in the domestic system to measure total energy consumption with high temporal resolution. In this chapter we use a deep learning method to disaggregate the low frequency energy signal generated directly by the new generation smart meters deployed in Italy, without the need of additional specific hardware. The performances obtained on two reference datasets are promising and demonstrate the applicability of the proposed approach.

This chapter is published as M. M. Manca, L. Massidda, "Deep learning based non-intrusive load monitoring with low resolution data from smart meters", in Communications in Applied and Industrial Mathematics, vol. 13 (1), pp. 39-56, 2022, doi: https://doi.org/10.2478/caim-2022-0004.

## 5.1 Introduction

Implementing an effective NILM technique for domestic and industrial use is a valuable resource for both the consumer and the utility. For a domestic user, for example, the knowledge of the consumption of household appliances over time promotes greater awareness of energy consumption, allowing an informed choice of consumption habits[111]. It can be used to rationally distribute and/or reduce load throughout the day, to effectively manage home automation systems, to participate in demand response programs [112, 113]. For utility companies on the other hand, the knowledge of the habits and needs of individual users allows the offer of personalized services, a better segmentation of users [111] and a better scheduling of supply [114, 115], greater accuracy in the prediction of consumption, and facilitates the implementation of demand response strategies.

Recent work on the subject can be divided into two main groups, distinguished by the sampling frequency of the signals used for analysis. The first group of these papers is based on the availability of power data measured at high sampling rates (from 50Hz to over kHz), obtained through dedicated hardware, with which to obtain the identification of switching events [116, 117]. The second group, which also represents the most current line of research, is based on lower frequency aggregate load analysis methods, obtained with dedicated hardware or directly from the smart meter with sampling periods ranging from 1s to 1 hour. This approach has been shown to be effective in obtaining, sometimes with reasonable accuracy, an estimate of the energy consumed by appliances with higher power demand even if used occasionally [118, 119, 120, 121].

NILM studies have tested both supervised and unsupervised approaches, depending on the information available and the predictive algorithm implemented. Supervised methods require a data set labeled with sub-metered devices, which is not always available. Unsupervised methods can be implemented without any prior knowledge of the environment, but the user is usually required to match the patterns identified by the algorithm to the appliances.

Early NILM studies were primarily based on hidden Markov models (HMM), which are typically used for probabilistic modeling of time series data [122, 123], these techniques are commonly inefficient when the number of devices increases and also suffer from high computational complexity [111].

A different approach is that of optimization methods, where the main idea is to find the optimal combination of the individual devices that make up the aggregated signal. The increased complexity resulting from a large number of devices and the loss of temporal continuity are two major drawbacks to using these techniques [124]. Thermal type loads, moreover, can be effectively disaggregated using Bayesian techniques [125].

Machine learning-based (non deep) approaches require manual feature construction using domain expert knowledge. These solutions have lower computational complexity than deep learning approaches and in some cases have yielded encouraging results. Numerous techniques for both regression and classification are used, such as K Nearest Neighbours [126], Support Vector Machines [127, 128], and especially tree-based methods that achieve performance comparable to newer deep learning techniques [129, 130, 131].

All machine learning methods require the practitioner to define descriptive features of the load that are then processed by regression or classification algorithms. In the deep learning approach, feature synthesis becomes an integral part of the machine learning process; in fact, the first layers of the deep network are used for processing the raw signal to obtain meaningful features that are then processed by subsequent layers.

A widely used approach is Convolutional Neural Networks (CNN), in which the first layers consist precisely of convolution layers that operate as filters on the raw signal [33, 132, 133, 134]. Subsequent layers use other types of layers such as pooling layers, normalization layers, and fully connected layers. The most common approach involves the use of 1d convolution modules that receive as input a time sequence of the aggregated load signal and return as output, the value of the load of one or more appliances, or the identification of its activation state, for a portion of the input sequence or even for just the midpoint of the interval.

Recurrent Neural Networks, are a type of deep learning architecture designed for temporal sequences, in which the basic concept is that information about previous states is part of the input for the next state. The most recent versions of this type of network are represented by the Gated Recurrent Unit (GRU) and Long Short Term Memory (LSTM) architectures. Several architectures with different complexity have been tested in the literature for NILM, in which usually the first layers are convolutional layers anyway, with performance comparable to that obtained with

convolutional networks [33, 34, 135, 136].

Another architecture studied in the literature is autoencoders (AE), an architecture consisting of an encoder, which compresses the input signal to a representation on a latent space and then reconstructs the signal itself using a portion of the network called a decoder. The latent space representation of the aggregate consumption signal can be used to extract information about the consumption of individual appliances [33, 137, 138].

More recent developments for processing sequential data include architectures that implement the attention mechanism, which, unlike recurrent architectures, uses all previous states of an encoder to construct the input for the decoder, in an overall sequence-to-sequence configuration [139, 140].

In [141] we used techniques from semantic image segmentation to build a deep learning model using convolutional networks for load disaggregation on a low frequency signal. The model proved to be very effective in disaggregating the loads of a house even in the case, more interesting from the application point of view, in which the neural network is applied to the load of a house not present in the dataset used for the training of the model. We have also experimented with network ensembling techniques to further improve the already good accuracy achieved by the network [142]. One of the peculiarities of the proposed model is the use of data with a reduced sampling rate (1 min) for both training and testing, in order to allow a possible application of the technique to measurements coming directly from the smart meter without the use of dedicated monitoring hardware.

Most of the techniques proposed in the literature in fact assume the availability of a measurement of total consumption of the user with a sampling rate higher than that generally obtainable from smart meters installed by the distributor, so they require a minimum of intrusiveness for the user as it is often necessary to equip the house with a measurement system of total instantaneous power dedicated to the disaggregation application.

Our interest is instead to evaluate the applicability of the methodology developed to the measurement signal obtainable from new generation smart meters being deployed throughout Europe. In particular, in this work we want to verify the applicability of the model for a *Chain 2* signal, the channel of communication on conveyed waves (PLC-C) adopted in the second generation smart meters distributed in Italy [143, 144]. The *Chain 2* signal combines a quarter-hourly consumption measure-

ment with a signal for load variation events; a measurement of the instantaneous load is sent every time it crosses pre-set threshold values. In this chapter, we will demonstrate that the approach we have developed is also applicable to this type of signal by allowing both user-side and utility-side load disaggregation without the need for dedicated measurement hardware. We will apply the neural network to two datasets in which the aggregated signal is filtered to reproduce a *Chain 2* type signal and we will verify the achievable performance. To our knowledge this is the first work to propose a disaggregation on the signal directly obtained from a commercial smart meter.

The chapter is structured as follows: the problem is formulated in Section 5.2.1, the methodology proposed for its solution is described in Section 5.2.2, Section 5.3 describes numerical experiments conducted on a reference dataset, the results of which are presented in Section 5.4; conclusions are drawn in Section 5.5 where some possible further developments of this research are presented.

## 5.2 Materials and methods

### 5.2.1 Problem formulation

If $y(t)$ represents the total active electrical power consumed at instant $t$, and we denote by $y_i(t)$ the active power absorbed by the appliance with index $i$ at the same instant, the total load can be expressed as the sum of the absorptions of the individual appliances and an unmeasured portion:

$$y(t) = \sum_{i=1}^{N} y_i(t) + e(t) \quad \forall t \in (0, T) \tag{5.1}$$

where $N$ is the number of appliances considered and $e(t)$ is the unidentified residual load.

The problem is to get the values of $y_i(t)$ when only the measure of $y(t)$ is known, that is to get an approximation of $F(y(t))$:

$$[y_1(t), y_2(t), \ldots, y_i(t), \ldots, y_N(t)] = F(y(t)) \tag{5.2}$$

where $F$ is the operator that, when applied to the total active power, returns $N$

distinct values that are the best estimate of the power absorbed by individual appliances. Note that, in general, $y_i(t)$ does not represent all electrical appliances, but a fraction of all those present in a house. The unknown term $e(t)$ thus takes into account loads due to unmonitored devices. The task of finding an approximation of the operator $F$ can be set up as a supervised learning problem when simultaneous measurements of aggregate load and consumption of individual appliances are available.

If, as in our case, we are primarily interested in cumulative consumption and activation times, the estimated consumption $haty_i(t)$ of individual devices can be approximated by a function that is constant over the activation period of the device:

$$\hat{y}_i(t) = p_i \hat{a}_i(t) \tag{5.3}$$

where $p_i$ is the average consumption of the appliance $i$ and the $\hat{a}_i(t)$ is an estimate of the state of activation of the individual appliance at the time $t$, which has a unit value if the appliance is in operation and zero value otherwise.

Therefore, the method we propose seeks to obtain the most accurate possible estimate of the state of activation of the appliances starting from the aggregate load,

$$[\hat{a}_1(t), \hat{a}_2(t), \ldots, \hat{a}_i(t), \ldots, \hat{a}_N(t)] = F_a(y(t)) \tag{5.4}$$

and obtains an estimate of consumption using Equation 5.3 after knowing the average nominal consumption of the appliances examined.

### 5.2.2 Methodology

In [141] we proposed a methodology to obtain the $F_a$ using a convolutional neural network architecture we called Temporal Pooling NILM (TP-NILM) an adaptation of the network called PSPNet (Pyramid Scene Parsing Network) proposed by Zhao et al. in [145] for the semantic segmentation of images.

The general scheme follows the classical approach to semantic image segmentation, where we have an encoder, which allows to increase the feature space of the signal at the cost of a reduction in its temporal resolution, and a decoder module that reconstructs an estimate of the activation state of the device at the same resolution as the original signal. In addition to these, there is a module called Temporal

Pooling that performs feature aggregation at different resolutions, generating a temporal context, which spans long periods without completely losing resolution in the signal description.

The network layout is shown in Figure 5.1.

Unlike other approaches presented in the literature, in this case the model tries to recognize only the activation periods of the appliances under consideration and does not try to reconstruct the detail of their absorption over time. The task is less related to the characteristics of the individual appliance, so a model trained on a few users can provide reasonable results even on a user, and on an appliance, that does not enter the training set, but that retains absorption characteristics typical of the appliance class to which it belongs.

The network weights are obtained by gradient descent optimization. The loss function is a binary Cross-Entropy applied to each of the output channels that measures the difference between the activations estimated by the network $\hat{a}_i(t)$ and the actual activations $a_i(i)$ for each device examined and for each instant of the time period under consideration. The network was implemented using the PyTorch library [146].

### 5.2.3 Chain 2 protocol

In 2019, the EU revised its energy policy framework through the Clean Energy for All Europeans package, marking a significant step toward implementing the Union's energy strategy. The importance of smart metering in delivering energy efficiency and empowering consumers by enabling their active participation in the electricity market is reaffirmed, also through the coupling of smart meters with consumer energy management systems. The directive states, among other things, that smart meters must send information about consumption to end-users: historical consumption and near real-time consumption [147].

The second generation Smart Meters distributed in Italy provide the possibility of communication with a user device through the new channel called *Chain 2* on power line (PLC-C), compliant with standard [148]. Some works have already described the potential of this technology and presented the communication properties between the in-home device and the smart meter [143].

The transmission of data occurs both synchronously and asynchronously; these types of transmission are used for communication between the smart meter and the

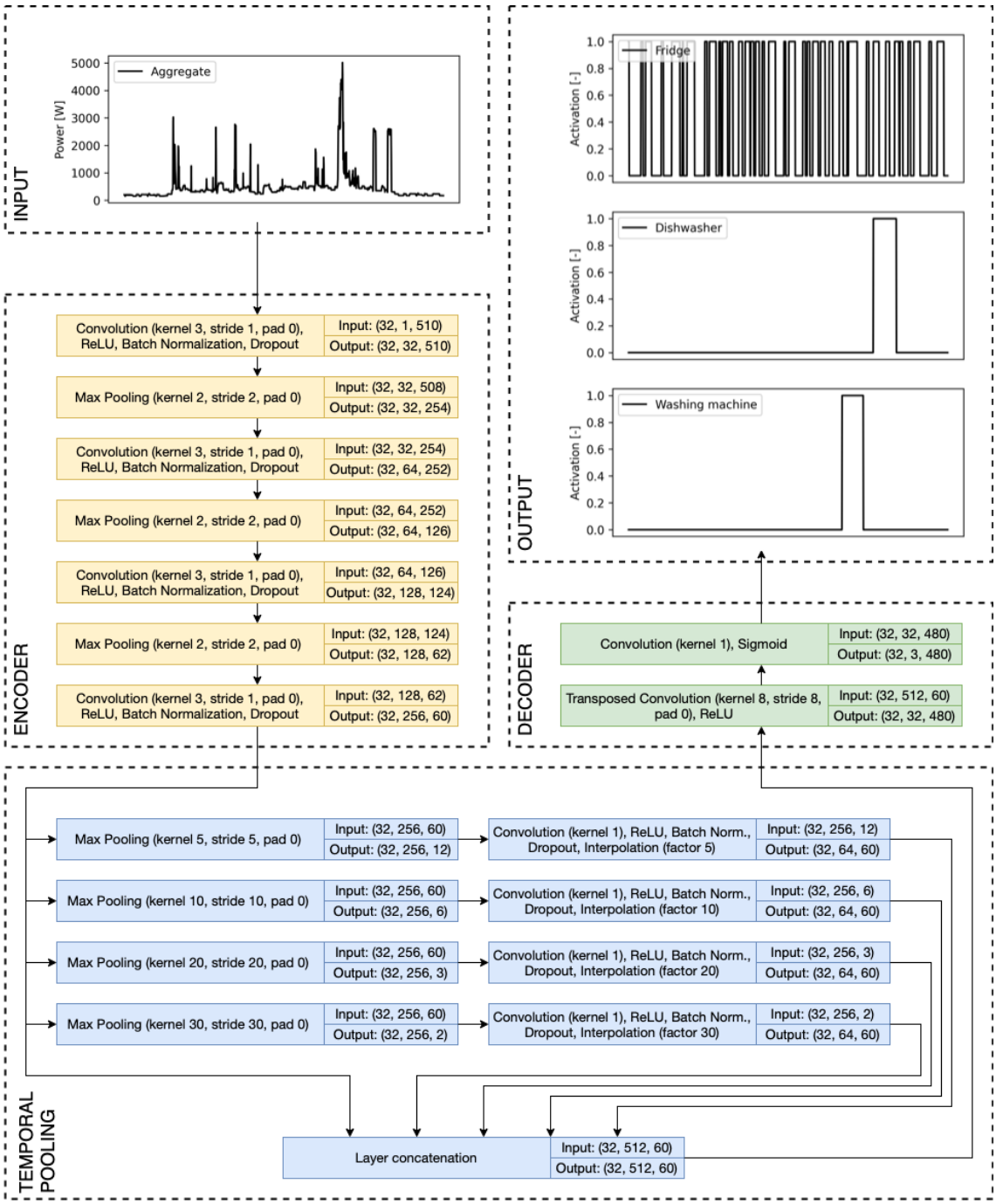


Figure 5.1: Outline of the network architecture used

in-home device enabling the user to monitor consumption. In the first type the power signal is sent every 15 minutes to the device, while in the second one the last active power sample is sent to the device when fixed thresholds are exceeded.

The operation of sample acquisition by the in-home device is described in detail in standard [149].

In Figure 5.2 the signal of the power measurements at 1 min intervals is shown together with the synchronous signal, sent every quarter of an hour, and with the asynchronous signal, sent each time the power crosses a pre-defined thresholds, uniformly distributed with a constant step equal to 500 W for the example. The data volume to be communicated is clearly lower with respect to the power measurements at a constant sampling rate.

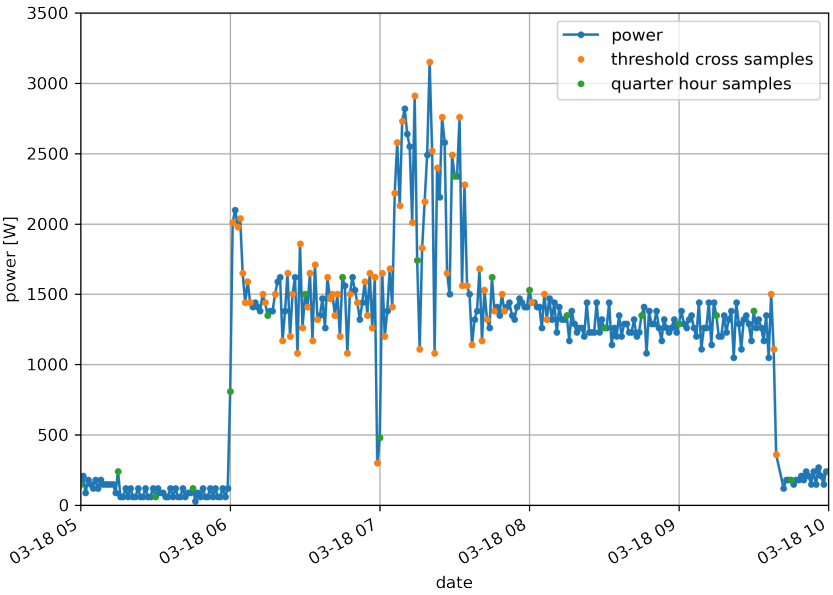


Figure 5.2: *Chain 2* filtering on load signal with power thresholds at 500 W intervals. Blue dots represent the load measurements with 1 min sampling rate, green dots are the power measurements at the constant rate of 15 min, orange dots are the power measurements sent to the user each time the power absorbed crosses a threshold, in this case the thresholds are set to [500, 1000, ..., 3500].

## 5.3 Experimental setup

We apply the TP-NILM architecture to two reference datasets, the UK-DALE dataset [150] and the REFIT dataset [151], to evaluate the accuracy in the dis-

aggregation of an always-on appliance, such as the refrigerator, and two household appliances whose activation can be deferred, such as dishwasher and washing machine, which are therefore interesting in a perspective of energy management. Th aggregate load for these dataset is available at a constant sampling rate. The time series is filtered to simulated a *Chain 2* signal. A time series with a constant sampling rate of 1 min is then reconstructed from the filtered signal and use as the input for the neural network. We then test the effectiveness of the disaggregation for different values of the preset thresholds for the *Chain 2* filter.

### 5.3.1 UK-DALE dataset

The UK-DALE dataset contains the aggregate and individual appliance signals from 5 different households in the United Kingdom. All aggregate loads were sampled at a frequency of 1Hz, unlike the individual appliance signals where the frequency was 1/6Hz.

The duration of the recordings is not the same in all the houses in the dataset, and the appliances in each house are also different. For this reason, after a careful analysis of the dataset, it was decided to use only houses 1, 2 and 5 for the purpose of the experiment being the only houses equipped with all the appliances considered.

### 5.3.2 REFIT dataset

Similarly to the previous case, the REFIT dataset contains the aggregate and individual appliance electricity consumption of 20 homes in United Kingdom. In this case the data is collected with a uniform frequency of 8 Hz for both aggregate and individual appliance consumption.

The timeline is not the same for all the houses in the dataset, but the records manage to cover a time span of two years. The richness of the data therefore allowed a considered choice of houses to be used for the experiments proposed in this work. A total of 15 houses were used, but unlike the previous case they were divided among the appliances of interest. The latter are however the same as those chosen for the UK-DALE case, i.e. fridge, dishwasher and washing machine.

### 5.3.3 Chain 2 filtering

The power measurements for the *Chain 2* signal useful for disaggregation purposes are of two types: an instantaneous load signal emitted at a constant rate every 15 min, and a signal in which the value of instantaneous power is transmitted whenever the power itself crosses fixed power values. For the second signal the possible values of the instantaneous load are divided into bins of fixed width (300 W is the default setting for the smart meter). Every time the instantaneous consumption belongs to a different bin than the last value communicated, the measurement of instantaneous power is transmitted.

**Algorithm 1** Calculate *Chain 2* signal

**Input:** aggregate, threshold
**Output:** filtered_aggregate
filtered_aggregate=aggregate
l=0
**for** i=0:length(filtered_aggregate) **do**
  **if** $\lfloor$filtered_aggregate[i] $\div$ threshold$\rfloor \neq$ l **then**
    l= $\lfloor$filtered_aggregate[i] $\div$ threshold$\rfloor$
  **else**
    filtered_aggregate[i]=*NaN*
  **end if**
**end for**
filtered_aggregate[*every 15 min*]=aggregate[*every 15 min*]

The aggregate consumption signal for households is filtered to reproduce a *Chain 2* type signal according to the Algorithm 1.

The threshold value, i.e., bin width, is a smart meter parameter configurable by the utility, and clearly influences the amount of data transmitted and the richness of the signal useful for a disaggregation algorithm.

Figures 5.3 and 5.4 show the data volume of aggregated, filtered and unfiltered, signals from some houses used in this work in the UK-DALE and REFIT case respectively. It is evident from the figures that in the cases examined, the threshold value has a limited influence on the overall volume of data, especially when compared to the amount of unfiltered signal data.

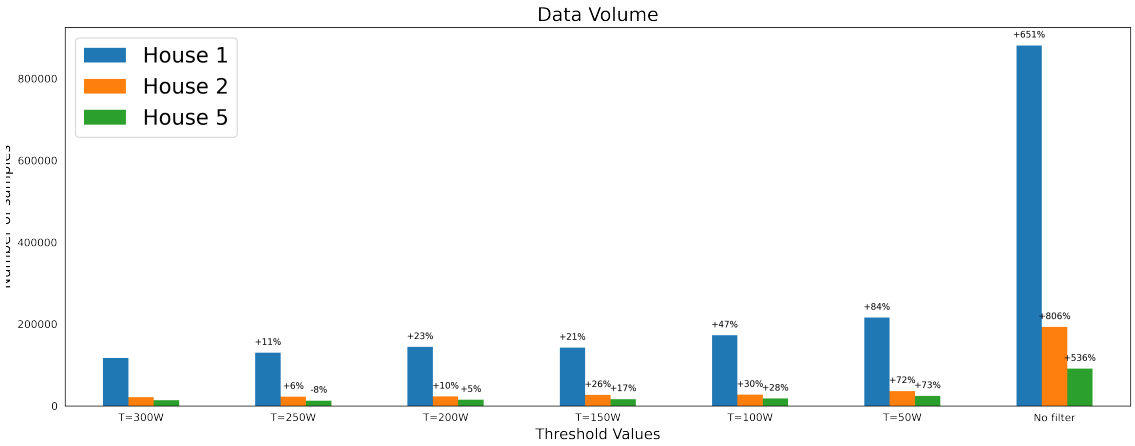


Figure 5.3: Number of samples for different threshold values of the *Chain 2* filter in UK-DALE dataset. A logaritmic scale was used and the percentages represent the increase in the number of samples compared to the number obtained with the default threshold value.

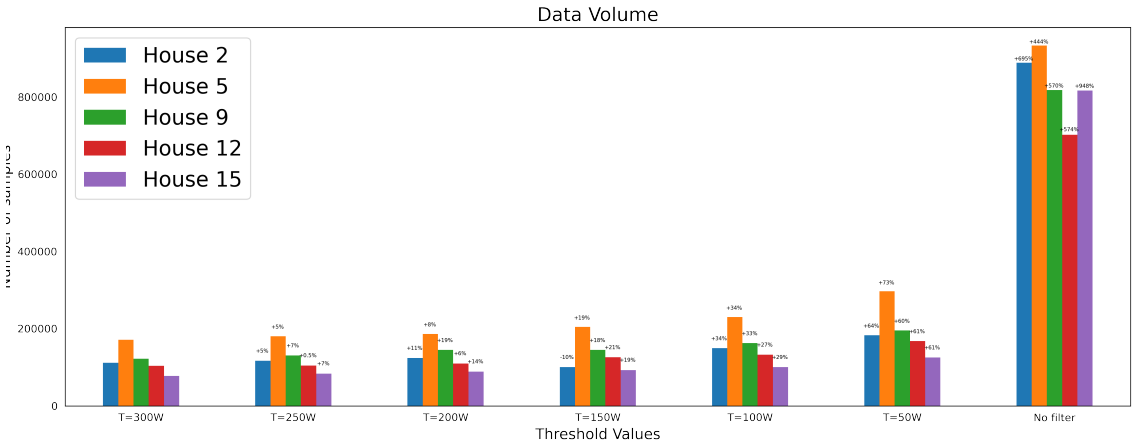


Figure 5.4: Number of samples for different threshold values of the *Chain 2* filter in REFIT dataset. A logaritmic scale was used and the percentages represent the increase in the number of samples compared to the number obtained with the default threshold value.

### 5.3.4 Preprocessing

Before training and using the neural network, the data from the two datasets were subjected to a pre-processing phase. As a first step, the power values of each appliance were extracted and sub-sampled at a frequency of 1/60 Hz, using the average power value in each time interval considered. Consequently, the aggregated signal was also resampled with the same approach and using the same frequency value.

The aggregated signal was then filtered to generate a *Chain 2* signal, and the output of the filtering was resampled over a period of 1 minute, propagating the last valid observation forward to the next valid one, in order to make it homogeneous and usable with the network.

The activation status of the individual appliances was then calulated with a threshold method, similarly to [33], since it appears to provide good results in terms of accuracy for both datasets [152]: each appliance is considered to be in operation when a certain threshold is exceeded for a fixed time interval, the Table 5.1 describes the thresholds and time intervals chosen, for UK-DALE dataset, taken from [33] and Table 5.2 the values for REFIT dataset as proposed in [152].

Table 5.1: Parameters used to derive the activation status of household appliances from measurements of the absorbed power for UK-DALE

| | **Fridge** | **Dishwasher** | **Washing Machine** |
|---|---|---|---|
| Max. power (W) | 300 | 2500 | 2500 |
| Power threshold (W) | 50 | 10 | 20 |
| OFF duration (min) | 0 | 3 | 30 |
| ON duration (min) | 1 | 30 | 30 |

Table 5.2: Parameters used to derive the activation status of household appliances from measurements of the absorbed power for REFIT

| | **Fridge** | **Dishwasher** | **Washing Machine** |
|---|---|---|---|
| Max. power (W) | 300 | 2500 | 2500 |
| Power threshold (W) | 20 | 30 | 20 |
| OFF duration (min) | 1 | 27 | 13 |
| ON duration (min) | 10 | 23 | 27 |

To facilitate convergence and lower the computational cost, the data were normalised by dividing each power value by a fixed value of 2000W. In addition, the time sequence of values was divided into windows of length 512 minutes, used as input to the neural network, which provides an output of 8 hours (480 minutes). The 32 minutes of difference between input and output constitute a leading and a trailing edge of 16 minutes each, due to the convolution filters for which no padding was applied.

Finally, in order for the input values to have zero mean, the average power value in each considered interval is subtracted from the signal values.

### 5.3.5 Training and testing

The techniques described so far have the main goal of predicting the activation status of three fixed appliances, fridge, washing machine and dishwasher, by knowing the aggregate signal of a household. For this purpose the Convolutional Neural Network shown in Figure 5.1 was trained.

To evaluate the effectiveness of the proposed methodology, the network was tested on the two datasets described above, UK-DALE and REFIT.

In both cases, the network was tested on houses not contained in the training set. In this way, it is possible to assess the generalisation capacity of the model presented, i.e. the ability of the network to disaggregate the consumption of appliances in house that do not belong to the training set and for which the time series of the appliance consumption are used for the sole purpose of performance assessment.

Regarding the UK-DALE dataset, the network was trained using the training portion of House 1 and House 5 and it was tested using the entire time series of House 2. Table 5.3 summarizes which portions of the dataset were used for the training, validation and testing phases.

Table 5.3: Training, Validation and Testing dataset composition for the UK-DALE dataset case.

| | **Training** | **Validation** | **Testing** |
|---|---|---|---|
| House 1 | 80 % | 20 % | - |
| House 2 | - | - | 100 % |
| House 5 | 80 % | 20 % | - |

Concerning the REFIT dataset, in accordance with [135], it was decided to use different houses for each appliance in the different phases of training, validation and testing. Table 5.4 summarizes which houses were chosen.

Note that during the training phase the input signal was not filtered with the *Chain 2* protocol.

In both cases the Adam optimisation algorithm was used to optimise the parameters of the neural network with a gradient descent approach, the learning rate was set at $5 \cdot 10^{-5}$ and the batch size at 32. In order to avoid the overfitting an early

Table 5.4: Training, Validation and Testing dataset composition for the REFIT dataset case.

| | **Training (80%) and Validation (20%)** | **Testing** |
|---|---|---|
| Fridge | 2, 5, 9, 12 | 15 |
| Dishwasher | 1, 3, 5, 6, 7, 9, 10, 11, 13, 15, 16 18, 20 | 2 |
| Washing Machine | 1, 2, 3, 5, 7, 8, 9, 10, 11, 13, 15, 16, 17, 18, 19, 20, 21 | 6 |

stopping strategy was adopted. The parameters described were not the only ones used, but they were the ones that gave good results in terms of convergence times and accuracy obtained.

### 5.3.6 Postprocessing

The output of the network is an array of values that varies in the range $(0, 1)$ and represents the probability that the appliance is turned on. The appliance is rated as switched on if the probability exceeds an arbitrated threshold which has been set at 0.5. An estimate of the power consumption related to the appliance is also calculated, it is a constant value equal to the average power consumption of each appliance multiplied by the duration of its operation.

### 5.3.7 Performance evaluation

The performance of the network was evaluated both for identifying the activation status of each device and for estimating power consumption. Several metrics were used for this purpose.

Let $a_i(t)$ be the activation state of the $i$-th appliance at time $t$, which is equal to 1 if the appliance is on or 0 if it is off, and let $y_i(t)$ the value of the power consumed at the same time. Then we can denote with $\hat{a}_i(t)$ and $\hat{y}_i(t)$ the model predictions of the quantities described above. The metrics for evaluating the model's predictions refer to a time series $t_k$ with $k \in [0, N_s)$.

We can also define by True Positive (TP) the number of times the model correctly predicts the activation state of an appliance when it is switched on, by True Negative (TN) the number of times the appliance is correctly evaluated as switched off. While we can define with False Positive (FP) the number of instances in which the appliance is evaluated as on when it is off and with False Negative (FN), on the contrary, the number of times it is evaluated as off despite being on.

Then we can define the following metrics.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{5.5}$$

$$Precision = \frac{TP}{TP + FP} \tag{5.6}$$

$$Recall = \frac{TP}{TP + FN} \tag{5.7}$$

$$F_1 = 2\frac{Precision \times Recall}{Precision + Recall} \tag{5.8}$$

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \tag{5.9}$$

As can be seen, the Equation 5.5 of Accuracy represents the ratio of the number of correctly evaluated instances to the total number of predicted instances. On the other hand Precision, in Equation 5.6, is defined as the ratio between TP and the total number of times the appliance is evaluated to be switched on, while Recall, as described in Equation 5.7, defines the ratio between the number of times the appliance is rated in operation and the total number of times it is actually in operation.

The F1 measurement defined in Equation 5.8 is the weighted average of Precision and Recall, varying in a range between $[0, 1]$ and high values of this score indicate a better ability to identify the state of the appliance. Finally the Matthews Correlation Coefficient is defined in Equation 5.9, whose values are in the range $[-1, 1]$. A value equal to 1 denotes an exact classification, 0 denotes a random prediction while a value equal to $-1$ indicates a totally wrong classification.

Note that the metrics defined so far have been used to assess performance in terms of predicting the activation state of an appliance, thus relating to a classification problem. Regarding the estimation of the energy consumed, the accuracy of the proposed methodology has been measured in terms of Mean Absolute Error (MAE) and Signal Aggregate Error (SAE), defined by the equations below.

$$MAE_i = \frac{1}{N_s} \sum_j \hat{y}_i(t_j) - y_i(t_j) \tag{5.10}$$

$$SAE_i = \frac{\sum_j \hat{y}_i(t_j) - \sum_j y_i(t_j)}{\sum_j y_i(t_j)} \tag{5.11}$$

MAE, defined in Equation 5.10, measures how much on average the estimated power deviates from the measured power at each instant, while SAE, defined in Equation 5.11, estimates the relative error of the predicted energy over the entire time interval considered.

The methodology we propose estimates instantaneous consumption by assuming a constant value during the appliance's activation cycle; this is a good approximation in the case of appliances with on/off operation such as the fridge, while it will result in higher values for MAE in the case of appliances with variable absorption during their activation, such as the dishwasher and washing machine. The MAE metric is of little significance to us; we report it to allow easier comparison with other methods proposed in the literature.

## 5.4 Results

The performance of the TP-NILM network was examined for the two UK-DALE and REFIT datasets, evaluating its accuracy both on the original signal and on the one filtered with the *Chain 2* protocol with different threshold values. The results for the three appliances considered for the UK-DALE dataset are shown in Table 5.5, Table 5.6 reports the performance obtained on the REFIT dataset.

In both datasets it is possible to see that the accuracy depends on the value of the threshold chosen for filtering. The maximum levels of accuracy are reached with the unfiltered signal, as expected, however filtering according to the *Chain 2* scheme still allows an effective disaggregation as long as the threshold values are not too high.

The results show that the proposed approach enables to obtain a very accurate disaggregation of the consumption of the appliances considered for both datasets, even using a low frequency sampling and a *Chain 2* type signal. The threshold, set for the event based component of the signal, is also a threshold for the appliances that can be disaggregated, an appliance that has a power consumption, when turned on, lower than or close to the threshold value cannot be disaggregated, because the consumption signal generated by it is filtered in the processing of the *Chain 2* signal.

In fact, it can be seen that the accuracy in the disaggregation of the fridge, falls for threshold values higher than 50W, since the absorption of the same, when the compressor is active, is slightly higher than 100W, typically. For the dishwasher, in both datasets, maximum accuracy can be obtained for threshold values up to 300W. For the washing machine, in both datasets, the maximum tolerable threshold is 100W; a threshold of 300W results in a significant deterioration of the performance of the proposed algorithm. Among the various metrics proposed we note the good values for the estimate of the activation state, the MAE has instead modest values, as it was deliberately chosen not to have an exact estimate of the instantaneous absorption as this information has little practical relevance. Much more important is the estimate of the total energy absorbed, whose accuracy is measured by the SAE has excellent values for the three appliances for thresholds up to 100W.

Therefore, the *Chain 2* signal can be directly used for load disaggregation as long as the chosen threshold is commensurate with the power level of the household appliance, and disaggregation can then be realized directly from smart meter measurements without the need to install additional measurement hardware.

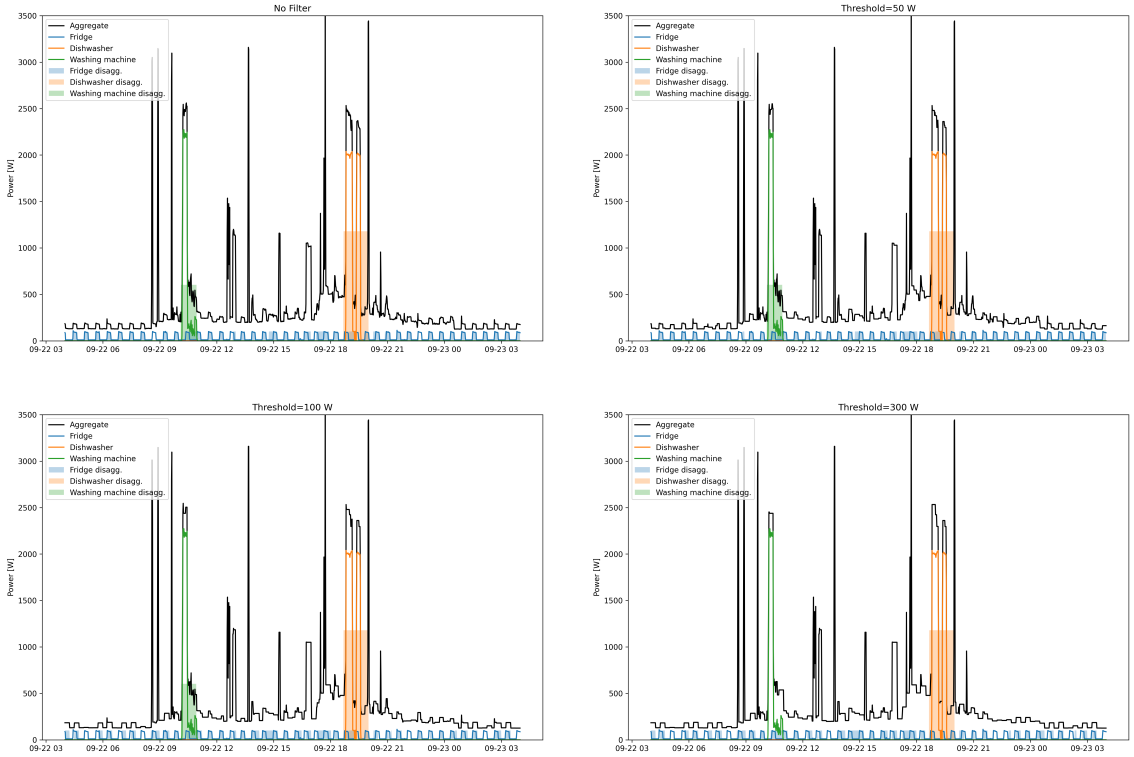


Figure 5.5: Example of load disaggregation for the UK-DALE dataset case. The graphs show the aggregate load and the load of each appliance as solid lines while the disaggregated estimate of the loads is represented by a shaded area. The first plot represents the estimate for the original (unfiltered) input signal while the other three are related to the threshold values chosen to perform the experiments.

Figure 5.5 shows an example application of the network for the UK-DALE case, the portion of the load signal was chosen so that the three appliances examined were simultaneously activated. The figures show the aggregate signals and actual consumption of the appliances superimposed on the estimate of activations and average consumption obtained with the proposed algorithm. As said, the algorithm allows to obtain a direct estimate of the activation status starting from the aggregate signal only, eventually filtered with the Chain 2 protocol. The estimate of the absorbed power is obtained, as said, considering a constant average power for the activation period, as is also evident from the graphical representation. In this example we can see the degradation of the disaggregation estimate for the refrigerator already for low thresholds, the good accuracy in the disaggregation of the dishwasher, independent of the threshold value adopted, and the sudden degradation in the estimation of the absorption of the washing machine for thresholds above 100W.

Tables 5.5 and 5.6 also report the results obtained from the most recent works in the literature that have examined the two datasets. This is not a fair comparison, as all the cited methods use the signal at the maximum sampling rate, which would not be possible to obtain from a smart meter, and therefore require additional measurement hardware for their application. Nevertheless, the performance of our proposed architecture is aligned with the performances of the best results in the recent literature.

In the comparison with the literature, our architecture obtains the worst performance according to the F1 metric in the case of the refrigerator, for both datasets. The reason is related precisely to the sampling of the signal, which we recall, even in the unfiltered case is limited to a rate of 1 min in our case, this signal does not allow the network to effectively detect the initial transient due to the compressor switching on. However, it should be noted that the performanche in terms of SAE for the refrigerator are nonetheless excellent. The value of F1 is lower than the best results in the literature also in the case of the washing machine in the REFIT dataset, which is compensated by the excellent performance in terms of SAE in this case as well.

## 5.5 Discussion

The results presented in the previous Section show that with a reasonable choice of filter parameters an accurate disaggregation is achieved with a significant reduction in the volume of data transmitted, thus realizing an effectively non-intrusive consumption monitoring, feasible both on user and utility side. These results also highlight how rich in information the electrical signal of consumption can be, and emphasise the need for special attention in the protection of sensitive information that can be derived from it. The limitations we see are typical of supervised approaches, which are highly dependent on the available data. The acquisition of labeled datasets is certainly expensive, but in principle the algorithm proves to be valid, and it is possible to think of its scalability when more complete datasets were available to facilitate the training of models.

Table 5.5: TP-NILM performances on House 2 of the *UK-DALE* dataset. NF (No Filter) represents network performances using the original aggregate signal.

| | **Prec.** | **Rec.** | **Acc.** | **F1** | **MCC** | **MAE** | **SAE** |
|---|---|---|---|---|---|---|---|
| **Fridge** | | | | | | | |
| Krystalakos et al. [34] GRU | 0.46 | 0.75 | 0.60 | 0.57 | - | 51 | 0.26 |
| Krystalakos et al. [34] s2p | 0.42 | 0.74 | 0.54 | 0.53 | - | 51 | 0.29 |
| Yue et al. [139] | - | - | 0.81 | 0.77 | - | 25.49 | - |
| Rafiq et al. [153] | - | - | - | 0.87 | 19.61 | 0.467 | |
| Zhou et al. [154] | 0.75 | 0.73 | - | 0.74 | - | 39.33 | - |
| Song et al. [135] | - | - | - | 0.94 | - | 22.35 | 0.12 |
| Piccialli et al. [140] | - | - | - | 0.87 | - | 13.24 | - |
| Puente et al. [155] | 0.97 | 0.96 | - | 0.97 | - | - | - |
| TP-NILM No filter | 0.87 | 0.89 | 0.90 | 0.89 | 0.79 | 17.35 | −0.05 |
| TP-NILM T=50W | 0.87 | 0.83 | 0.89 | 0.85 | 0.76 | 18.36 | −0.05 |
| TP-NILM T=100W | 0.70 | 0.70 | 0.77 | 0.70 | 0.52 | 27.67 | 0.01 |
| TP-NILM T=150W | 0.78 | 0.76 | 0.83 | 0.77 | 0.64 | 23.07 | −0.03 |
| TP-NILM T=200W. | 0.67 | 0.68 | 0.75 | 0.67 | 0.47 | 29.51 | 0.02 |
| TP-NILM T=250W | 0.67 | 0.66 | 0.75 | 0.67 | 0.46 | 29.68 | −0.01 |
| TP-NILM T=300W | 0.64 | 0.63 | 0.73 | 0.63 | 0.41 | 31.60 | −0.01 |
| **Dishwasher** | | | | | | | |
| Krystalakos et al. [34] GRU | 0.62 | 0.42 | 0.97 | 0.50 | - | 24 | 0.07 |
| Krystalakos et al. [34] s2p | 0.47 | 0.43 | 0.96 | 0.45 | - | 21 | 0.07 |
| Yue et al. [139] | - | - | 0.97 | 0.67 | - | 16.18 | - |
| Rafiq et al. [153] | - | - | - | 0.81 | 15.27 | 0.323 | |
| Zhou et al. [154] | 0.88 | 0.86 | - | 0.87 | - | 118.1 | - |
| Song et al. [135] | - | - | - | 0.87 | - | 20.95 | 0.69 |
| Piccialli et al. [140] | - | - | - | 0.72 | - | 7.26 | - |
| Puente et al. [155] | 0.73 | 0.90 | - | 0.81 | - | - | - |
| TP-NILM No filter | 0.82 | 0.85 | 0.99 | 0.83 | 0.83 | 30.95 | 0.04 |
| TP-NILM T=50W | 0.81 | 0.87 | 0.99 | 0.84 | 0.84 | 30.89 | 0.07 |
| TP-NILM T=100W | 0.81 | 0.85 | 0.99 | 0.83 | 0.82 | 31.08 | 0.05 |
| TP-NILM T=150W | 0.82 | 0.86 | 0.99 | 0.84 | 0.83 | 30.82 | 0.05 |
| TP-NILM T=200W | 0.81 | 0.85 | 0.99 | 0.83 | 0.82 | 30.56 | 0.05 |
| TP-NILM T=250W | 0.79 | 0.87 | 0.99 | 0.83 | 0.83 | 31.63 | 0.10 |
| TP-NILM T=300W | 0.80 | 0.82 | 0.99 | 0.81 | 0.80 | 31.45 | 0.02 |
| **Washing machine** | | | | | | | |
| Krystalakos et al. [34] GRU | 0.22 | 0.54 | 0.96 | 0.31 | - | 30 | 0.58 |
| Krystalakos et al. [34] s2p | 0.26 | 0.55 | 0.97 | 0.35 | - | 17 | 0.28 |
| Yue et al. [139] | - | - | 0.97 | 0.33 | - | 6.98 | - |
| Rafiq et al. [153] | - | - | - | 0.77 | 14.42 | 0.512 | |
| Zhou et al. [154] | 0.74 | 0.99 | - | 0.85 | - | 55.90 | - |
| Song et al. [135] | - | - | - | 0.88 | - | 12.27 | 0.26 |
| Piccialli et al. [140] | - | - | - | 0.69 | - | 6.57 | - |
| Puente et al. [155] | 0.43 | 0.40 | - | 0.41 | - | - | - |
| TP-NILM No filter | 0.76 | 0.92 | 0.99 | 0.83 | 0.84 | 9.57 | 0.21 |
| TP-NILM T=50W | 0.78 | 0.87 | 0.99 | 0.82 | 0.82 | 9.24 | 0.12 |
| TP-NILM T=100W | 0.82 | 0.81 | 0.99 | 0.82 | 0.82 | 8.64 | −0.01 |
| TP-NILM T=150W | 0.89 | 0.65 | 0.99 | 0.75 | 0.75 | 7.88 | −0.28 |
| TP-NILM T=200W | 0.88 | 0.53 | 0.99 | 0.66 | 0.68 | 7.89 | −0.40 |
| TP-NILM T=250W | 0.91 | 0.41 | 0.99 | 0.56 | 0.61 | 7.43 | −0.55 |
| TP-NILM T=300W | 0.92 | 0.20 | 0.99 | 0.32 | 0.43 | 7.15 | −0.78 |

Table 5.6: TP-NILM performances on House 15 (*Fridge* case), House 2 (*Dishwasher* case) and House 6 (*Washing Machine* case) of the *REFIT* dataset.

| | **Prec.** | **Rec.** | **Acc.** | **F1** | **MCC** | **MAE** | **SAE** |
|---|---|---|---|---|---|---|---|
| **Fridge** | | | | | | | |
| Song et al. [135] | - | - | - | 0.95 | - | 20.15 | 0.13 |
| Pan et al. [156] | - | - | - | - | - | 16.77 | 0.10 |
| Murray et al. [132] CNN | - | - | 0.77 | 0.93 | - | 8.56 | - |
| Murray et al. [132] GRU | - | - | 0.64 | 0.85 | - | 13.30 | - |
| D'Incecco et al. [133] CNN | - | - | - | - | - | 20.02 | 0.33 |
| TP-NILM No filter | 0.81 | 0.79 | 0.89 | 0.80 | 0.73 | 9.37 | −0.02 |
| TP-NILM T=50W | 0.69 | 0.71 | 0.83 | 0.70 | 0.58 | 14.05 | 0.03 |
| TP-NILM T=100W | 0.61 | 0.64 | 0.78 | 0.62 | 0.48 | 17.33 | 0.06 |
| TP-NILM T=150W | 0.61 | 0.64 | 0.79 | 0.63 | 0.48 | 17.21 | 0.05 |
| TP-NILM T=200W. | 0.59 | 0.63 | 0.78 | 0.61 | 0.45 | 18.15 | 0.07 |
| TP-NILM T=250W | 0.59 | 0.63 | 0.78 | 0.61 | 0.45 | 18.06 | 0.06 |
| TP-NILM T=300W | 0.57 | 0.60 | 0.76 | 0.58 | 0.42 | 19.08 | 0.06 |
| **Dishwasher** | | | | | | | |
| | **Prec.** | **Rec.** | **Acc.** | **F1** | **MCC** | **MAE** | **SAE** |
| Jiang et al. [134] | - | - | - | 0.60 | - | 17.66 | - |
| Cimen et al. [136] | - | - | - | 0.78 | - | 13.35 | - |
| Song et al. [135] | - | - | - | 0.88 | - | 12.34 | 0.60 |
| Pan et al. [156] | - | - | - | - | - | 4.80 | 0.17 |
| Murray et al. [132] CNN | - | - | 0.83 | 0.82 | - | 82.74 | - |
| Murray et al. [132] GRU | - | - | 0.85 | 0.82 | - | 73.53 | - |
| D'Incecco et al. [133] CNN | - | - | - | - | - | 12.26 | 0.26 |
| TP-NILM No filter | 0.79 | 0.86 | 0.98 | 0.82 | 0.81 | 61.91 | 0.09 |
| TP-NILM T=50W | 0.79 | 0.86 | 0.98 | 0.82 | 0.81 | 62.18 | 0.09 |
| TP-NILM T=100W | 0.78 | 0.86 | 0.98 | 0.82 | 0.81 | 62.39 | 0.10 |
| TP-NILM T=150W | 0.78 | 0.86 | 0.98 | 0.82 | 0.81 | 62.70 | 0.11 |
| TP-NILM T=200W | 0.78 | 0.86 | 0.98 | 0.82 | 0.81 | 62.97 | 0.11 |
| TP-NILM T=250W | 0.78 | 0.86 | 0.98 | 0.82 | 0.81 | 62.98 | 0.12 |
| TP-NILM T=300W | 0.77 | 0.87 | 0.98 | 0.82 | 0.80 | 63.38 | 0.12 |
| **Washing machine** | | | | | | | |
| | **Prec.** | **Rec.** | **Acc.** | **F1** | **MCC** | **MAE** | **SAE** |
| Jiang et al. [134] | - | - | - | 0.75 | - | 3.91 | - |
| Cimen et al. [136] | - | - | - | 0.94 | - | 8.70 | - |
| Song et al. [135] | - | - | - | 0.89 | - | 8.88 | 0.22 |
| Pan et al. [156] | - | - | - | - | - | 5.88 | 0.13 |
| Murray et al. [132] CNN | - | - | 0.72 | 0.79 | - | 71.99 | - |
| Murray et al. [132] GRU | - | - | 0.69 | 0.86 | - | 79.33 | - |
| D'Incecco et al. [133] CNN | - | - | - | - | - | 16.85 | 2.61 |
| TP-NILM No filter | 0.79 | 0.89 | 0.99 | 0.84 | 0.84 | 8.54 | 0.13 |
| TP-NILM T=50W | 0.87 | 0.89 | 0.99 | 0.88 | 0.87 | 7.96 | 0.03 |
| TP-NILM T=100W | 0.90 | 0.87 | 0.99 | 0.89 | 0.88 | 7.66 | −0.03 |
| TP-NILM T=150W | 0.91 | 0.79 | 0.99 | 0.85 | 0.85 | 7.38 | −0.13 |
| TP-NILM T=200W | 0.92 | 0.66 | 0.99 | 0.77 | 0.78 | 6.99 | −0.28 |
| TP-NILM T=250W | 0.93 | 0.54 | 0.99 | 0.69 | 0.71 | 6.74 | −0.42 |
| TP-NILM T=300W | 0.95 | 0.40 | 0.99 | 0.57 | 0.62 | 6.36 | −0.58 |

# Chapter 6

# Appliance Recognition with combined single- and multi-label approaches

The problem of appliance recognition is one of the most relevant issues in the field of Non-Intrusive-Load-Monitoring; its importance has led, in recent years, to the development of innovative techniques to try to solve it. The use of methods such as V-I trajectory, Fryze Theory Decomposition and Weighted Recurrence Graph have proved effective in recognising both single (*Single Label*) and multiple active appliances (*Multi Label*). This chapter presents a new way of approaching the problem by unifying *Single Label* and *Multi Label* learning paradigms. The proposed approach exploits feature extraction techniques which allow the detection of both activated/deactivated appliances and all active appliances given aggregate current signal. We evaluate the proposed approach on a PLAID dataset. The obtained results indicate combining single-label and mult-label learning strategies for appliance recognition provides improved classification results with an F-score of 0.91.

This chapter is published as M. M. Manca, A. Faustine, L. Pereira "Appliance Recognition with Combined Single- and Multi-label Approaches", in Proceedings of the 9th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation (BuildSys '22). Association for Computing Machinery, New York, NY, USA, 388–392. https://doi.org/10.1145/3563357.3566153

## 6.1 Introduction

The applicance recognition problem, described in the Introduction of this thesis, can be approached from two perspectives: in the *Single-Label* approach, the goal is to recognize appliances using the individual device's on/off signals; on the other hand, in the *Multi-Label* approach, the aggregate signal is used to understand which devices are activated and which are not. [37]-[38]-[39]

A major challenge is to solve the two problems simultaneously; many works offer interesting solutions by addressing these with innovative feature extraction and classification techniques, but in a separate way. It has been shown how using Weighted Recurrence Graphs can help increase the performance of classification algorithms about appliance recognition [41], as well as using pre-processing techniques based on Fryze theory decomposition [40].

In order to provide a more accurate classification, deep neural networks are increasingly being used in the NILM [33]-[141]; their effectiveness is, however, undoubtedly constrained by the availability of rich data. For this reason, high-frequency signal data are often used, allowing for more efficient training of this network. [157]-[158]

The main focus of this chapter is to open up a new challenge, namely to tackle the two problems mentioned above using a single algorithm capable of combining the two tasks in such a way that the information sharing of the two models can increase performance. So far, a combined algorithm has been created and tested that would homogeneously share information from the two models. Still, it would be of great interest to create one that would allow the exchange of information between the two in an optimized manner. The proposed method was tested on the publicly available Plug-Load Appliance Identification Dataset (PLAID) [159].

## 6.2 Methodology

The proposed work aims to detect which appliance $m$ has been switched on or off and simultaneously estimate the state $s_m(t)$ of all $M$ appliances from the aggregate current and voltage signal $x(t)$, for $m = 1, \ldots, M$. The aggregate signal can be

described by 6.1:

$$x(t) = \sum_{m=1}^{M} y_m(t) \cdot s_m(t) + \sigma(t) \tag{6.1}$$

where $y_m(t)$ represents the individual appliance measurements and $\sigma(\cdot)$ the noise contribution.

### 6.2.1 Problem formulation

The problem can be formulated as follows: let $\mathbf{A} \in \mathbb{R}^{T \times d_1}$ be the set of the input features obtained from the measurements of the switched-on or switched-off appliance, computed from the aggregate signal like the difference between the before and after the event; let $\mathbf{X} \in \mathbb{R}^{T \times d_2}$ be the aggregated signal values after an event; $\mathbf{M} = \{1, \ldots, M\}$ the set of indices representing each appliance; $\mathbf{Y} \in \mathbb{R}^{T \times M}$ the set of each appliance measurement composing the aggregate signal and $\mathbf{s}(t) = [s_1(t), \ldots, s_M(t)]$ the vector containing the state of each appliance at time $t \in \{1, \ldots, T\}$, where $s_j(t) \in \{0, 1\}$. Then, given a dataset $\mathbf{D} = \{\mathbf{a}(t), m, \mathbf{x}(t), \mathbf{s}(t) | \mathbf{a}(t) \in \mathbf{A}, m \in \mathbf{M}, \mathbf{x}(t) \in \mathbf{X}, t = 1, \ldots, T\}$, the aim is to train a classifier that can predict at the same time $t$ which appliance $m$ is activated and the state vector $\mathbf{s}(t)$ using $\mathbf{a}(t)$, $\mathbf{x}(t)$. It is important to note that whenever at instant $t^*$ one of the $m$ devices is switched on or off, the current and voltage consumption are added or subtracted to the aggregate consumption $x(t^*)$. Hence, this work assumes that appropriate information sharing during the training phase of both models will allow each task to be solved more efficiently. We will refer to the first task as *Single Label* problem and the second one as *Multi Label* problem.

### 6.2.2 Pre-processing and feature extraction

In this work, features derived from the current and the voltage of switching appliances on/off were used in the *Single Label* case; features derived from the aggregate current and aggregate voltage were used in the *Multi Label* case. In both cases, these are high-frequency measurements and one-cycle steady-state current $i(t)$ and voltage $v(t)$ signals were used, extracted as in [41], [160].

With the application of Fryze Theory Decomposition, which has been shown to improve performances because it can provide a distinctive feature for classification [161], these features are subsequently used to calculate the decomposed current

features, i.e. given the signals $i(t)$ and $v(t)$, it is possible to decompose $i(t)$ into two components via 6.2:

$$i(t) = i_a(t) + i_f(t) \tag{6.2}$$

where $i_a(t)$ and $i_f(t)$ represent respectively the active and non-active components of the current.

Once this is done, the Piece-wise Aggregate Approximation is used to reduce the dimensionality of the feature to a predetermined $w$, set to 50 in our work.

To the current signals thus obtained the Euclidian distance function is then applied:

$$d_{jk} = \|i(t)_j - i(t)_k\|_2$$

In this way, the relationship between active and non-active current is measured. These distances can also be interpreted as the entries of the distance similarity matrix $D \in \mathbb{R}^{w,w}$ for the points $i(t)_1, \ldots, i(t)_w$:

$$D = \begin{bmatrix} 0 & d_{12} & \cdots & d_{1w} \\ d_{21} & 0 & \cdots & d_{2w} \\ \vdots & \vdots & \ddots & \vdots \\ d_{w1} & d_{w2} & \cdots & 0 \end{bmatrix}$$

Finally, this matrix is used to calculate the Weighted Recurrence Graph matrix $WRG = [r_{jk}]$ with:

$$r_{jk} = \begin{cases} \delta & \text{if } \tau > \delta \\ \tau & \text{otherwise} \end{cases}$$

where $\tau = \left\lfloor \frac{d_{jk}}{\epsilon} \right\rfloor$ and the parameters $\delta$ and $\epsilon$ are chosen, in our case, both equal to 10.

This matrix is then used as input for the classifier, which in our case is a Convolutional Neural Network. The described algorithm is summarized in Figure 6.1.

### 6.2.3 Combined model

The main part of this work is to give an idea of how the two problems can be approached from a multi-task perspective using a single algorithm. The idea was, therefore, to train two different classifiers, one for the single-label case and one for

the multi-label case, which would share the information of the respective features in the learning phase. We propose two CNNs with the same structure for both problems, the only difference being the output size. Each classifier consist of four-stage CNN layer each with 16, 32, 64 and 128 feature maps, $1 \times 1$ strides and each layer uses a $3 \times 3$ filter size. The four layers are followed by a batch normalization layer and a Relu activation function, and the last layer is followed by an adaptive average pooling layer with an output size $1 \times 1$. The output layer consists of two fully connected layers with hidden size, 1024 and $M$, in the *Single Label* case, or $2M$, in the *Multi Label* case, where $M$ is the number of available appliances. For both cases, the final predictions are obtained by applying the logarithmic softmax.

In the first case, the algorithm's output can be interpreted as a vector whose components represent each appliance's logarithm of the probability of activation or deactivation. Similarly, in the second case, the output represents two vectors containing the logarithm of the probability of each device being switched on or off.

To allow the parameters of the two models to share information during the learning phase, backpropagation was used to optimise a combined objective function defined by the equation:

$$\mathcal{L}_{comb}(\mathbf{m}, \hat{\mathbf{m}}, \mathbf{s}, \hat{\mathbf{s}}) = \mathcal{L}_{sl}(\mathbf{m}, \hat{\mathbf{m}}) + \mathcal{L}_{ml}(\mathbf{s}, \hat{\mathbf{s}}) \tag{6.3}$$

where $\mathcal{L}_{sl}$, $\mathcal{L}_{ml}$ represent the loss function in the *Single-Label* and in the *Multi-Label* case respectively. For both cases, the Negative Log-Likelihood Loss function is used [162].

The mini batch Stochastic Gradient Descent with a momentum of 0.9 and lerning rate of 0.001 is used to train the single-label model and an Adam optimizer with the same learning rate and betas of $(0.9, 0.98)$ is used to train the multi-label model. The combined model is then trained for 300 iterations using a batch size equal to 16 and to avoid the over-fitting we used an early-stopping with patience where the training phase is stopped once the validation performances don't change after 100 iterations.

The use of a combined loss to allow simultaneous training between the two models can be improved by introducing some penalty parameters to be added during the learning phase. Although the results of our approach, described in the Section 6.4, are interesting, the purpose of this work is to propose a new approach to the problem

and offer an idea for an improvable solution.

## 6.3 Evaluation

### 6.3.1 Dataset

The proposed algorithm was tested on the PLAID dataset. This public dataset contains records, collected at high frequency (30kHz), measuring the individual current and voltage of 17 devices and the aggregate current and voltage of the combination of 13 of these. The data was collected at 65 locations in Pittsburgh, Pennsylvania (USA). In order to test our algorithm, 1046 activation and deactivation current and voltage measurements were extracted from 12 different individual appliances. This set of 2092 samples was used for the single-label part of our model. Each of these samples corresponds to the aggregated current and voltage measurements; when one or more devices are switched on or off consecutively, resulting in 2092 aggregated current and voltage samples used for the multi-label part of the model. The Figure 6.2 shows an example of the signals used, namely, a *Fan* current activation cycle, a *Vacuum* current activation cycle and the two appliances aggregated current cycle are illustrated.

The Figure 6.3 summarises the distribution of extracted appliances and the number of active appliances in the 2092 samples.

### 6.3.2 Performance metrics

In order to evaluate the performance of our model, several metrics were used. The four main metrics used are Recall, Precision, and F1 score [163, 164]. These metrics were calculated to monitor the performance of the individual appliances in the two models to understand where one model could perform better than the other and study a way to combine the two strategies

$$\text{Recall}_{\text{macro}} = \frac{1}{M}\sum_{i=1}^{M}\text{Recall}^{(i)} \tag{6.4}$$

$$\text{Precision}_{\text{macro}} = \frac{1}{M}\sum_{i=1}^{M}\text{Precision}^{(i)} \tag{6.5}$$

$$\mathrm{F1}_{\mathrm{macro}} = \frac{1}{M} \sum_{i=1}^{M} \mathrm{F1}^{(i)} \tag{6.6}$$

where $\mathrm{Recall}^{(i)}$, $\mathrm{Precision}^{(i)}$, $\mathrm{F1}^{(i)}$ are the metrics computed for the $i$-th appliance and $M$ is the number of total appliances.

### 6.3.3 Training and testing

Our approach is benchmarked using the multi-label stratified 3-fold Cross Validation; in this way, in addition, to avoid overfitting, the label percentages in each fold are fixed.

Although the classifiers' training phase takes place simultaneously and optimises the same loss function, it should be emphasised that the single-label classifier is trained using only the features extracted from the current and voltage signals of the appliance switched on or off. In contrast, the multi-label classifier uses the features from the aggregated signals.

During the test phase, predictions are computed considering the maximum logarithmic softmax output, and 30% of the data for each fold is used. The predictions of each fold are then merged into a single sequence to assess average performances.

## 6.4 Results

The Figure 6.4 represents the performances for each appliance of the two simultaneously trained models, and it contains the metrics described in the previous Section.

As can be seen by looking at the values of the metrics, for some appliances, there is a difference in performance between the two tasks (for instance, *Fan* or *Blender*); our idea is that by properly combining the two classification strategies the model with the better performance can help increase the performance of the other.

Although not comparable, the two classifiers have similar good characteristics from the point of view of general performance. The Table 6.1 summarises the general performance.

A miss-classification error analysis was also carried out to understand whether the switching on or off of some appliance could cause errors not common to both

Table 6.1: Generalisation performances between *Single Label* model and *Multi Label* model on the PLAID dataset.

| | **Single Label Model** | **Multi Label Model** |
|---|---|---|
| $\mathbf{Recall_{macro}}$ | 0.913 | 0.889 |
| $\mathbf{Precision_{macro}}$ | 0.913 | 0.944 |
| $\mathbf{F1_{macro}}$ | 0.910 | 0.915 |

models and, in this way, to think about how the model that makes mistakes could learn from the model that does not make mistakes.

Figure 6.5 illustrates the number of these errors, they are divided into errors committed only by one model and common errors; more precisely, errors committed only by the single-label model but not by the multi-label model (*Only SingleLabel Errors*), those committed by the multi-label model but not by the single-label model (*Only MultiLabel Errors*), and those committed by both algorithms (*Common Errors*) are counted.

It can be seen from the graph that in the case of the *Laptop*, for example, the multi-label model makes many more errors than the single-label model; appropriate information sharing between the two models could lead to a reduction in the number of errors.

The confusion matrix related to the predictions of the single-label model, illustrated in Figure 6.6, points out, for example, that a good number of errors made by the classifier are related to the *Air Conditioner* and also that most of these are related to deactivation events. In this case, the multi-label model could help the single-label classification.

## 6.5 Discussion

In this chapter, we proposed an approach to address two types of appliance recognition problems simultaneously; single and multi-label appliance recognition. The work aims to motivate the search for a model capable of sharing information between two classifiers under training to improve appliance recognition.

Our approach, albeit in a primitive form, was tested on the public dataset PLAID to prove its effectiveness. In fact, from the results obtained, it seems that the proposed method can offer an excellent basis to address and solve the problem. In

addition, we think the proposed approach could also be used to improve the event detection algorithm, which will likely pave the way toward end-to-end NILM.

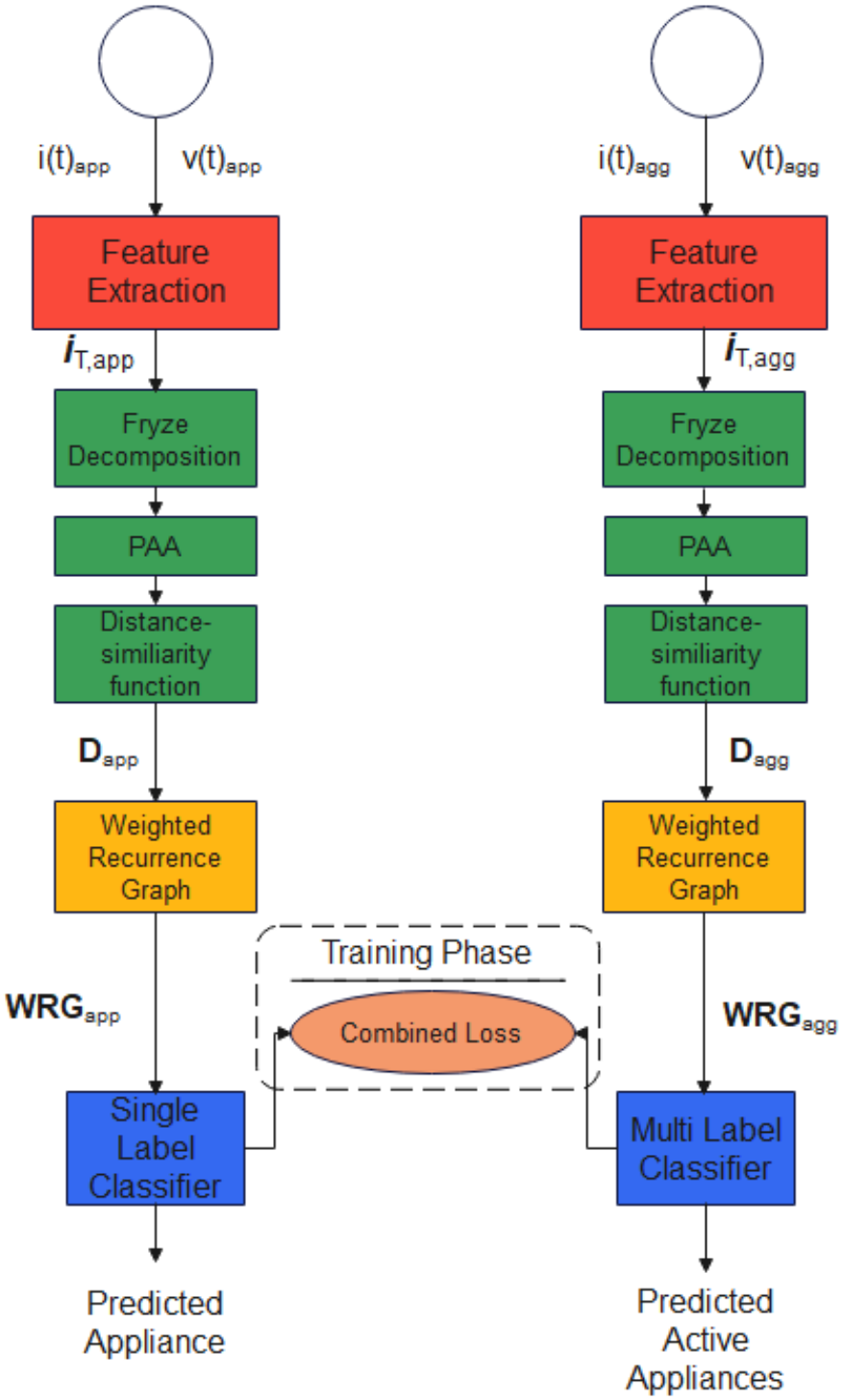


Figure 6.1: Block diagram of the proposed algorithm. The dotted block shows the training phase of the two classifiers, in which they learn by using the same combined loss function.

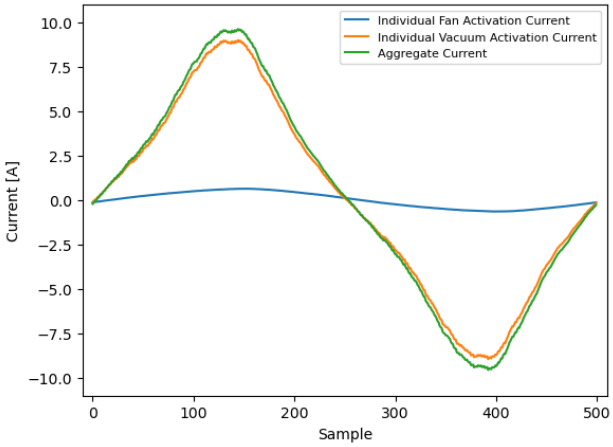


Figure 6.2: Extracted activation currents for two different events and the extracted aggregate current for these two events.

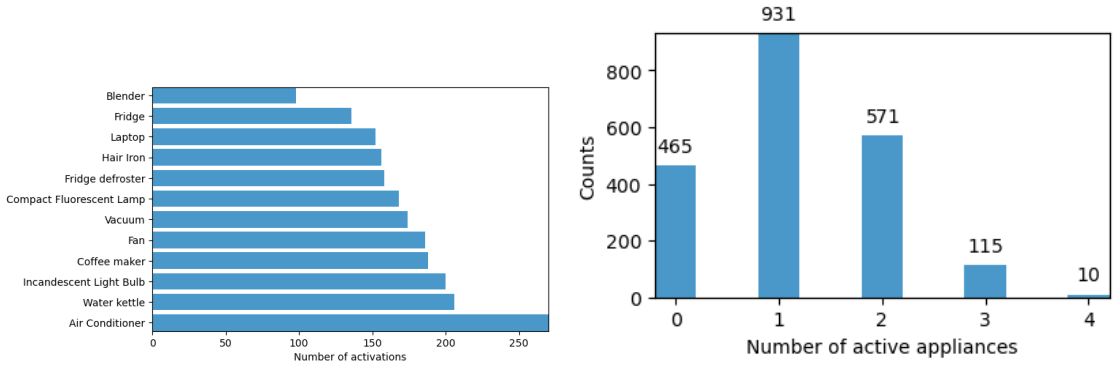


Figure 6.3: Appliances distribution on the extracted 2092 events (on the left) and the active appliances distribution (on the right).

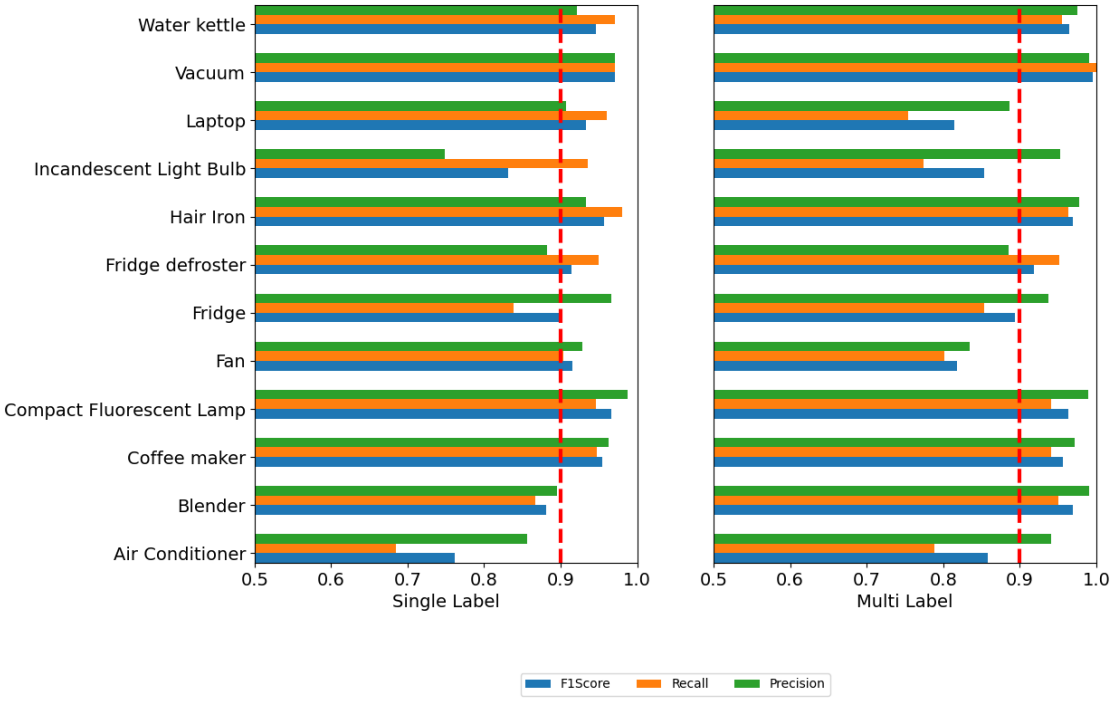


Figure 6.4: Per-appliance F1 score, Recall and Precision on the PLAID dataset. *Single Label* (left), *Multi Label* (right).

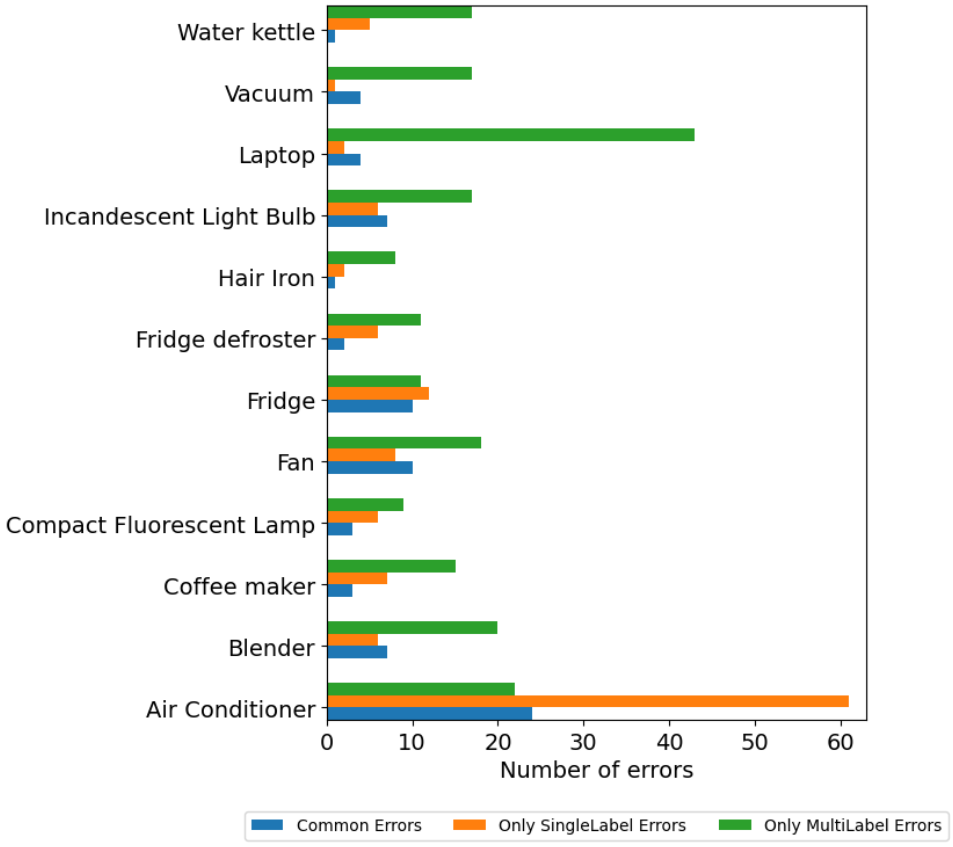


Figure 6.5: Number of incorrect predictions (during the testing phase) divided into errors committed by the *Single Label* model only, the *Multi Label* model only and those committed by both models (*Common Errors)*.

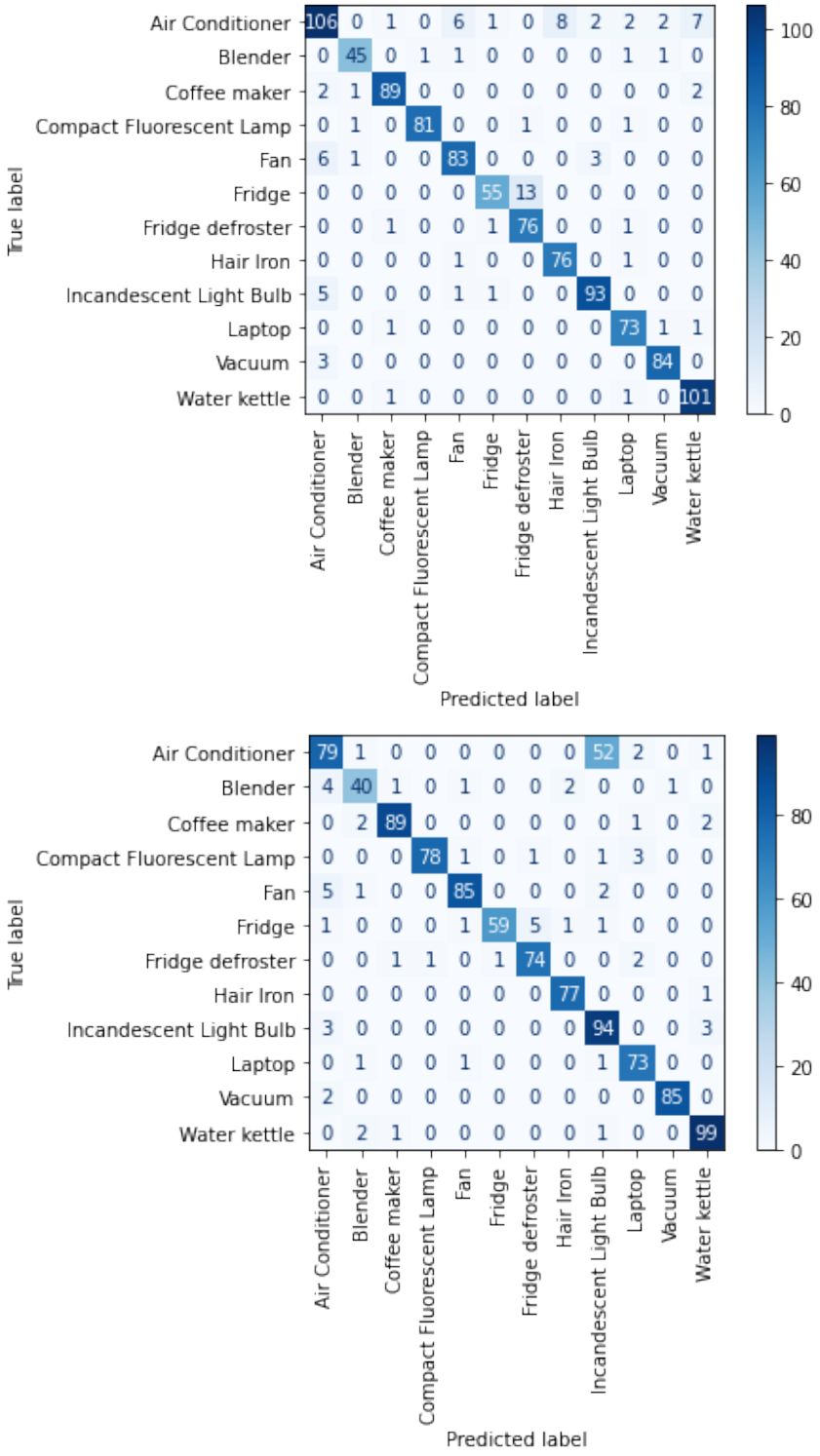


Figure 6.6: Confusion matrices of *Single Label* model predictions. On the top only the predictions for activation events are considered, on the bottom only the predictions for deactivation events.

# Chapter 7

# Conclusion

Recent advances in mobile sensor technology have opened up growing opportunities for HAR applications in a variety of fields, from personal wellness to health monitoring. However, as data collection becomes easier and easier, fully exploiting the available data to build reliable models for activity recognition remains challenging. Further, when dealing with wearable devices, the issues of limited resources and energy consumption cannot be neglected, posing additional requirements in terms of efficiency of the induced models. In such a perspective, feature selection may have an important role since it can significantly reduce the data dimensionality by removing irrelevant and noisy features.

In the Chapter 3, we have explored the potential of feature selection in the HAR field, leveraging five public mobile sensor datasets with heterogeneous characteristics and relying on a methodological framework that considers both the predictive power and the stability of the selected features subsets (i.e., their robustness to perturbations in the input data). Ten different selection methods have been comparatively evaluated, revealing the suitability of univariate ranking approaches, such as *Chi Squared* and *Information Gain*, as well as of some multivariate approaches, such as *ReliefF*, which exhibit a quite good trade-off between recognition performance and selection stability. Encompassing different levels of dimensionality reduction, our analysis has shown that the original number of features can be reduced to a very large extent in all the considered benchmarks, without any worsening of performance.

Such a large comparative study gives evidence of the potential benefits of systematically exploiting feature selection when inducing HAR models, also providing

methodological insights into how to evaluate the robustness of the selection outcome and choose the optimal level of dimensionality reduction. This line of research is worthy of further investigation, given the lack of such kind of comparative studies in this field.

As future work, our study will be enlarged to better investigate the extent to which the behavior and the stability pattern of a given selection algorithm may depend on the underlying feature extraction strategy (i.e., the type and number of features extracted for each sensor). Furthermore, the ranking-based selection framework here adopted will be compared with different and more sophisticated methodological approaches, including hybrid techniques that exploit different heuristics at different stages of the selection process and ensemble techniques that suitably combine the outcome of different selectors.

In the Chapter 4, we have presented a preliminary context-aware system for energy consumption forecasting in smart homes. Our system acquires context data from heterogeneous smart homes sensors. Those data are processed by artificial intelligence algorithms to recognize current activities, actions, and tasks. We collect the raw and inferred data using a sliding window approach, and we extract feature vectors including power consumption information. Those data are shared in anonymous form to train machine learning algorithms, which are used for energy forecasting tasks. An experimental evaluation with real world data shows that our approach is promising.

The work presented can be extended and improved in several directions. Feature selection methods could be applied to reduce overfitting and reduce the computational cost of our methods. Activity forecasting methods could be introduced to improve the system accuracy. Experiments with long-term observations in different smart homes, as well as comparison with other approaches, should be performed to better assess the effectiveness of our system.

The contribution presented in Chapter 5 addressed the disaggregation of the loads of some household appliances of interest, starting from the signal that is directly provided by the second generation smart meters deployed in Italy, without the need of additional hardware for the measurement of the aggregated load, therefore with a fully non-intrusive procedure. We applied a deep learning methodology based on convolutional neural networks to two reference datasets, for which we filtered the aggregate load signal simulating the *Chain 2* protocol, used in smart meters.

Clearly, extensions and additions can be applied to the proposed work. In particular, it might be interesting to modify the trained classifier so that it is able to predict the activation state of a larger number of appliances in order to increase the applicability of the work to a real case. Furthermore, it would be worth trying to use a different dataset between the training set and the test set, which would assess the generalisation capacity of the techniques adopted.

Finally, the work presented in the Chapter 6 is intended to encourage the combined approach of two interrelated problems. One of the strategies to be applied for future work is to consider a linear combination of the two loss functions instead of the sum to train the classifiers. The coefficients of the combination could be considered as weights or penalty factors or not less than trainable parameters. Such approaches, moreover, could be validated using different datasets in addition to the one proposed, e.g. a dataset that allows the use of Time-Series cross-validation and enables an approach much closer to the real case. A post-processing step could be helpful to improve the performance of the proposed model by considering the predictions of one of the two classifiers and checking that they are somehow compatible with the predictions of the other classifier.

In conclusion, this thesis aims to pave the way for the development of new intelligent systems that combine human activity recognition and electricity consumption monitoring to help users avoid waste and improve their quality of life and the world around them. The combination of these two fields of research can be used by energy communities to better manage the production and consumption of energy from renewable sources, as predicting consumption and recognising a user's activities in real time would allow them to optimise incoming electrical loads. Furthermore, the results obtained could help the development of systems capable of advising the exchange of energy between users of the same energy community. Certainly, the methods presented in this work can be refined and tested on combined datasets; an important research direction to be addressed would be to combine human activity recognition with electrical signal disaggregation in a single algorithmic system. Indeed, activity recognition would add useful information for monitoring electrical power. On the other hand, power monitoring would also allow users to make immediate future activity and consumption decisions. For this reason, the results of this thesis can be considered for strategic developments in the field of decision making for future activities, whether energy-related or not. An interesting opportunity is of-

fered by the crossover between machine learning and game theory; the contribution that the two approaches can make to the problem would be remarkable.

# Appendix A

# Feature description

In this appendix, our aim is to give an overview of the statistical measures that were computed to build the feature vectors of the considered datasets (i.e., *COSAR*, *HAR*, *HAR_AAL*, *HAPT*, *DSA*).

All the measures are listed in Table A.1, with the corresponding formula. Note that the third column of the table cites the datasets that use the measure and, in brackets, the number of times it was calculated.

In each formula of the table, the variable $x$ represents the signal. Depending on the dataset used and the feature calculated, this signal $x$ can be understood as an acceleration, an inclination, an angular velocity, a magnitude signal, or a Jerk signal (that is the first time derivative of the acceleration).

As mentioned in Section 3.4, the signals were segmented into sliding windows for feature extraction; specifically, all the measures reported in Table A.1 are calculated on windows containing a number of observations denoted by $M$.

It should also be considered that the sensors used in the experiments (accelerometers, magnetometers, and gyroscopes) acquire the three spatial components of each signal; thus, in some measures, $x$ can represent the component on the X, Y or Z axes of the considered signal. Other features, such as correlation and covariance, are calculated using two or more components on the axes; therefore, in the formulas it has been necessary to differentiate these components, indicating with $x_1$, $x_2$ and $x_3$ the components on the X, Y and Z axes respectively. Some other features, such as median and percentiles, are calculated using the window signal values sorted in ascending order, so we have indicated the ordered values differently using $\tilde{x}$.

As already pointed out in Section 3.4, the feature vectors were computed in both

Table A.1: Statistical measures used for feature extraction.

| Measure | Formula | Datasets |
|---|---|---|
| Mean | $\bar{x} = \frac{\sum_{k=1}^{M} x(i)}{M}$ | *COSAR* (14), *HAR* (46), *HAR_ALL* (46), *HAPT* (46), *DSA* (45) |
| Standard Deviation | $s = \sqrt{\frac{\sum_{i=1}^{M}(x(i)-\bar{x})^2}{M}}$ | *HAR* (33), *HAR_ALL* (33), *HAPT* (33) |
| Variance | $s^2 = \frac{\sum_{i=1}^{M}(x(i)-\bar{x})^2}{M}$ | *COSAR* (14), *DSA* (45) |
| Covariance | $Cov_{j,k} = \sum_{i=1}^{M}(x_j(i)-\bar{x}_j)(x_k(i)-\bar{x}_k)$ | *COSAR* (18) |
| Correlation | $Cor_{j,k} = \frac{\sum_{i=1}^{M}(x_j(i)-\bar{x}_j)(x_k(i)-\bar{x}_k)}{\sqrt{\sum_{i=1}^{M}(x_j(i)-\bar{x}_j)^2(x_k(i)-\bar{x}_k)^2}}$ | *COSAR* (18), *HAR* (15), *HAR_ALL* (15), *HAPT* (15) |
| Median | $m = \begin{cases} \tilde{x}(\frac{M+1}{2}) \text{ if } M \text{ is odd} \\ \frac{\tilde{x}(\frac{M}{2})+\tilde{x}(\frac{M}{2}+1)}{2} \text{ if } M \text{ is even} \end{cases}$ | *COSAR* (14), *HAR* (33), *HAR_ALL* (33), *HAPT* (33) |
| Min/Max Values | $\min = \min_{i=1,\dots,M}\{x(i)\}$; $\max = \max_{i=1,\dots,M}\{x(i)\}$ | *COSAR* (40), *HAR* (79), *HAR_ALL* (79), *HAPT* (79), *DSA* (90) |
| 90th Percentile | $p_{90} = \tilde{x}(0.90 \cdot M)$ | *COSAR* (14) |
| Harmonic Mean | $h = \frac{M}{\sum_{i=1}^{M}\frac{1}{x(i)}}$ | *COSAR* (2) |
| Signal Energy | $SE_j = \sum_{i=1}^{M}\lvert X_j(i)\rvert^2$ | *HAR* (159), *HAR_ALL* (159), *HAPT* (159) |
| Entropy | $E_j = -\sum_{i=1}^{M}\frac{\lvert X_j(i)\rvert^2}{\sum_{k=1}^{M}\lvert X_j(k)\rvert^2}\log_2\left(\frac{\lvert X_j(i)\rvert^2}{\sum_{k=1}^{M}\lvert X_j(k)\rvert^2}\right)$ | *HAR* (33), *HAR_ALL* (33), *HAPT* (33) |
| Skewness | $Sk = \frac{\sum_{i=1}^{M}\frac{1}{M}(x(i)-\bar{x})^3}{\sqrt{\left(\sum_{i=1}^{M}\frac{1}{M}(x(i)-\bar{x})^2\right)^3}}$ | *HAR* (13), *HAR_ALL* (13), *HAPT* (13), *DSA* (45) |
| Kurtosis | $Kur = \frac{\sum_{i=1}^{M}\frac{1}{M}(x(i)-\bar{x})^4}{\left(\sum_{i=1}^{M}\frac{1}{M}(x(i)-\bar{x})^2\right)^2} - 3$ | *COSAR* (14), *HAR* (13), *HAR_ALL* (13), *HAPT* (13), *DSA* (45) |
| Autoregressive Coefficients | If $x_j(i) = \sum_{i=1}^{p}\phi_k x_j(i-k) + \epsilon(i)$ is the Autoregressive Model, then $\phi_k$ are the **Autoregressive Coefficients** | *HAR* (80), *HAR_ALL* (80), *HAPT* (80) |
| Interquartile Range | $IqR = Q_3 - Q_1$, $Q_1$ and $Q_3$ are the first and third quartile. | *HAR* (33), *HAR_ALL* (33), *HAPT* (33) |
| Signal Magnitude Area | $SMA = \frac{1}{M}\sum_{i=1}^{M}\lvert x_1(i)\rvert + \lvert x_2(i)\rvert + \lvert x_3(i)\rvert$ | *HAR* (17), *HAR_ALL* (17), *HAPT* (17) |
| Angle | $A(i) = \arccos\frac{x_3(i)}{\sqrt{x_1(i)^2+x_2(i)^2+x_3(i)^2}}$ | *HAR* (7), *HAR_ALL* (7), *HAPT* (7) |
| Autocorrelation | $Auto(k) = \frac{1}{M-k}\sum_{i=0}^{M-k-1}(x(i)-\bar{x})(x(i-k)-\bar{x})$ | *DSA* (450) |
| Peaks | Let be $S_{DFT}(k) = \sum_{i=1}^{M}x(i)e^{-j2\pi ki/M}$ the $k$-th element of the 1-D $M$-point Discrete Fourier Transform, then **Peaks** are the maximum five Fourier peaks. | *DSA* (225) |
| Frequency Peaks | The frequency values that correspond to the Fourier **Peaks** | *DSA* (225) |

the time domain and the frequency domain; some measures in fact, e.g. Entropy, have been calculated in the frequency domain; consequently, in Table A.1, the frequency signal is indicated with the capital letter $X$, to distinguish it from the signal $x$ in the time domain.

# Bibliography


[1] E. De-La-Hoz-Franco, P. Ariza-Colpas, J. M. Quero, and M. Espinilla, "Sensor-based datasets for human activity recognition - a systematic review of literature," *IEEE Access*, vol. 6, pp. 59 192–59 210, 2018.

[2] K. D. al., "Activity recognition based on inertial sensors for ambient assisted living," in *2016 19th International Conference on Information Fusion (FUSION*, 2016, pp. 371–378.

[3] J. Qi, P. Yang, A. Waraich, Z. Deng, Y. Zhao, and Y. Yang, "Examining sensor-based physical activity recognition and monitoring for healthcare using internet of things: A systematic review," *Journal of Biomedical Informatics*, vol. 87, pp. 138–153, 2018.

[4] K. de Barbaro, "Automated sensing of daily activity: A new lens into development," *Developmental psychobiology*, vol. 61, no. 3, pp. 444–464, 2019.

[5] E. Khodabandehloo, D. Riboni, and A. Alimohammadi, "Healthxai: Collaborative and explainable ai for supporting early diagnosis of cognitive decline," *Future Generation Computer Systems*, vol. 116, pp. 168–189, 2021.

[6] W. S. Lima, E. Souto, K. El-Khatib, R. Jalali, and J. Gama, "Human activity recognition using inertial sensors in a smartphone: An overview," *Sensors*, vol. 19, p. 3213, 2019.

[7] F. Demrozi, G. Pravadelli, A. Bihorac, and P. Rashidi, "Human activity recognition using inertial, physiological and environmental sensors: A comprehensive survey," *IEEE Access*, vol. 8, pp. 10 816–21 083, 2020.

[8] B. Nguyen, Y. Coelho, T. Bastos, and S. Krishnan, "Trends in human activity recognition with focus on machine learning and power requirements," *Machine Learning with Applications*, vol. 5, no. 100072, 2021.

[9] A. Ahmad, M. Hassan, M. Abdullah, H. Rahman, F. Hussin, H. Abdullah, and R. Saidur, "A review on applications of ann and svm for building electrical energy consumption forecasting," *Renewable and Sustainable Energy Reviews*, vol. 33, pp. 102–109, 2014. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1364032114000914

[10] A. Camara, W. Feixing, and L. Xiuqin, "Energy consumption forecasting using seasonal arima with artificial neural networks models," *International Journal of Business and Management*, vol. 11, no. 5, p. 231, 2016.

[11] N. Dey, S. Fong, W. Song, and K. Cho, "Forecasting energy consumption from smart home sensor network by deep learning," in *International conference on smart trends for information technology and computer communications*. Springer, 2017, pp. 255–265.

[12] M. Gams, I. Y.-H. Gu, A. Härmä, A. Muñoz, and V. Tam, "Artificial intelligence and ambient intelligence," *Journal of Ambient Intelligence and Smart Environments*, vol. 11, no. 1, pp. 71–86, 2019.

[13] G. W. Hart, "Nonintrusive appliance load monitoring," *Proceedings of the IEEE*, vol. 80, no. 12, pp. 1870–1891, 1992.

[14] S. M. Tabatabaei, S. Dick, and W. Xu, "Toward non-intrusive load monitoring via multi-label classification," *IEEE Transactions on Smart Grid*, vol. 8, no. 1, pp. 26–40, 2016.

[15] D. Li and S. Dick, "Whole-house non-intrusive appliance load monitoring via multi-label classification," in *2016 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2016, pp. 2749–2755.

[16] M. Kahl, T. Kriechbaumer, A. U. Haq, and H.-A. Jacobsen, "Appliance classification across multiple high frequency energy datasets," in *2017 IEEE International Conference on Smart Grid Communications (SmartGridComm)*. IEEE, 2017, pp. 147–152.

[17] G. Chetty, M. White, and F. Akther, "Smart phone based data mining for human activity recognition, procedia computer science," *vol.*, vol. 46, pp. 1181–1187, 2015.

[18] P. Gupta and T. Dallas, "Feature selection and activity recognition system using a single triaxial accelerometer," *in IEEE Transactions on Biomedical Engineering*, vol. 61, no. 6, pp. 1780–1786, 2014.

[19] N. Ahmed, J. Rafiq, and M. Islam, "Enhanced human activity recognition based on smartphone sensor data using hybrid feature selection model," *Sensors*, vol. 20, no. 1, p. 317, 2020.

[20] F. Amjad, M. H. Khan, M. A. Nisar, M. S. Farid, and M. Grzegorzek, "A comparative study of feature selection approaches for human activity recognition using multimodal sensory data," *Sensors*, vol. 21, no. 7, 2021. [Online]. Available: https://www.mdpi.com/1424-8220/21/7/2368

[21] J. Chong, P. Tjurin, M. Niemelä, T. Jämsä, and V. Farrahi, "Machine-learning models for activity class prediction: A comparative study of feature selection and classification algorithms," *Gait and Posture*, vol. 89, pp. 45–53, 2021.

[22] I. Amezzane, Y. Fakhri, M. E. Aroussi, and M. Bakhouya, "Analysis and effect of feature selection over smartphone-based dataset for human activity recognition," in *Emerging Technologies for Developing Countries*, F. Belqasmi, H. Harroud, M. Agueh, R. Dssouli, and F. Kamoun, Eds. Cham: Springer International Publishing, 2018, pp. 214–219.

[23] J. Suto, S. Oniga, and P. P. Sitar, "Comparison of wrapper and filter feature selection algorithms on human activity recognition," in *6th International Conference on Computers Communications and Control (ICCCC)*, 2016, pp. 124–129.

[24] R. Chen, C. Dewi, and S. Huang, "Selecting critical features for data classification based on machine learning methods," *J Big Data*, vol. 7, p. 52, 2020.

[25] S. K. Bashar, A. A. Fahim, and K. H. Chon, "Smartphone based human activity recognition with feature selection and dense neural network," *42nd*

*Annual International Conference of the IEEE Engineering in Medicine Biology Society (EMBC)*, pp. 5888–5891, 2020.

[26] R. Guha, A. H. Khan, P. K. Singh, R. Sarkar, and D. Bhattacharjee, "Cga: A new feature selection model for visual human action recognition," *Neural Computing and Applications*, vol. 33, pp. 5267–5286, 2021.

[27] B. Völker, A. Reinhardt, A. Faustine, and L. Pereira, "Watt's up at home? smart meter data analytics from a consumer-centric perspective," *Energies*, vol. 14, no. 3, 2021. [Online]. Available: https://www.mdpi.com/1996-1073/14/3/719

[28] H. Dong, J. Zhu, S. Li, W. Wu, H. Zhu, and J. Fan, "Short-term residential household reactive power forecasting considering active power demand via deep transformer sequence-to-sequence networks," *Applied Energy*, vol. 329, p. 120281, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0306261922015380

[29] E. U. Haq, X. Lyu, Y. Jia, M. Hua, and F. Ahmad, "Forecasting household electric appliances consumption and peak demand based on hybrid machine learning approach," *Energy Reports*, vol. 6, pp. 1099–1105, 2020, 2020 The 7th International Conference on Power and Energy Systems Engineering. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2352484720314967

[30] H. Shi, M. Xu, and R. Li, "Deep learning for household load forecasting—a novel pooling deep rnn," *IEEE Transactions on Smart Grid*, vol. 9, no. 5, pp. 5271–5280, 2018.

[31] S. Ai, A. Chakravorty, and C. Rong, "Household power demand prediction using evolutionary ensemble neural network pool with multiple network structures," *Sensors*, vol. 19, no. 3, 2019. [Online]. Available: https://www.mdpi.com/1424-8220/19/3/721

[32] A. Harell, S. Makonin, and I. V. Bajić, "Wavenilm: A causal neural network for power disaggregation from the complex power signal," *arXiv preprint arXiv:1902.08736*, 2019.

[33] J. Kelly and W. Knottenbelt, "Neural nilm: Deep neural networks applied to energy disaggregation," in *Proceedings of the 2nd ACM International Conference on Embedded Systems for Energy-Efficient Built Environments*, ACM. ACM, 2015, pp. 55–64.

[34] O. Krystalakos, C. Nalmpantis, and D. Vrakas, "Sliding window approach for online energy disaggregation using artificial neural networks," in *Proceedings of the 10th Hellenic Conference on Artificial Intelligence*. ACM, 2018, pp. 1–6.

[35] E. Gomes and L. Pereira, "Pb-nilm: Pinball guided deep non-intrusive load monitoring," *IEEE Access*, vol. 8, pp. 48 386–48 398, 2020.

[36] F. Culière, L. Leduc, and A. Belikov, "Bayesian model of electrical heating disaggregation," in *Proceedings of the 5th International Workshop on Non-Intrusive Load Monitoring*, 2020, pp. 25–29.

[37] D. de Paiva Penha and A. R. G. Castro, "Home appliance identification for nilm systems based on deep neural networks," *Int. J. Artif. Intell. Appl*, vol. 9, pp. 69–80, 2018.

[38] M. A. A. Rehmani, S. Aslam, S. R. Tito, S. Soltic, P. Nieuwoudt, N. Pandey, and M. D. Ahmed, "Power profile and thresholding assisted multi-label nilm classification," *Energies*, vol. 14, no. 22, p. 7609, 2021.

[39] C. Nalmpantis and D. Vrakas, "On time series representations for multi-label nilm," *Neural Computing and Applications*, vol. 32, no. 23, pp. 17 275–17 290, 2020.

[40] A. Faustine, L. Pereira, and C. Klemenjak, "Adaptive Weighted Recurrence Graphs for Appliance Recognition in Non-Intrusive Load Monitoring," *IEEE Transactions on Smart Grid*, pp. 1–1, 2020.

[41] A. Faustine and L. Pereira, "Improved appliance classification in non-intrusive load monitoring using weighted recurrence graph and convolutional neural networks," *Energies*, vol. 13, no. 13, p. 3374, 2020.

[42] J. R. Kwapisz, G. M. Weiss, and S. A. Moore, "Activity recognition using cell phone accelerometers," *ACM SIGKDD Explor. Newsl.*, vol. 12, no. 2, pp. 74–82, 2011.

[43] J. L. Reyes-Ortiz, L. Oneto, A. Samà, X. Parra, and D. Anguita, "Transition-aware human activity recognition using smartphones," *Neurocomputing*, vol. 171, pp. 754–767, 2016.

[44] D. Riboni and C. Bettini, "Cosar: hybrid reasoning for context-aware activity recognition," *Personal and Ubiquitous Computing*, vol. 15, no. 3, pp. 271–289, 2011.

[45] M. M. Hassan, M. Uddin, A. Mohamed, and A. Almogren, "A robust human activity recognition system using smartphone sensors and deep learning," *Future Generation Computer Systems*, vol. 81, pp. 307–313, 2018.

[46] O. Lara and M. Labrador, "A survey on human activity recognition using wearable sensors," *IEEE Communications Surveys and Tutorials*, vol. 15, no. 3, 2013.

[47] A. Ferrari, D. Micucci, M. Mobilio, and P. Napoletano, "Trends in human activity recognition using smartphones," *J Reliable Intell Environ*, vol. 7, pp. 189–213, 2021.

[48] N. A. Capela, E. D. Lemaire, and N. Baddour, "Feature selection for wearable smartphone-based human activity recognition with able bodied, elderly, and stroke patients," *PLoS ONE*, vol. 10, no. 4, 2015.

[49] A. Aguileta, R. Brena, O. Mayora, E. Molino-Minero-Re, and L. Trejo, "Multi-sensor fusion for activity recognition - a survey," *Sensors*, vol. 19, p. 3808, 2019.

[50] A. Reiss, G. Hendeby, and D. Stricker, "A competitive approach for human activity recognition on smartphones," *Eur Symp. Artif. Neural Networks, Comput. Intell. Mach. Learn*, pp. 455–460, 2013.

[51] K. Rahim, I. Elamvazuthi, L. Izhar, and G. Capi, "Classification of human daily activities using ensemble methods based on smartphone inertial sensors," *Sensors*, vol. 18, no. 12, p. 4132, 2018.

[52] R. Zhu, Z. Xiao, Y. Li, M. Yang, Y. Tan, L. Zhou, S. Lin, and H. Wen, "Efficient human activity recognition solving the confusing activities via deep ensemble learning," *IEEE Access*, vol. 7, pp. 75 490–75 499, 2019.

[53] A. Loddo, B. Pes, and D. Riboni, "Feature selection in mobile activity recognition: A comparative study," *22nd IEEE International Conference on Mobile Data Management (MDM)*, pp. 181–186, 2021.

[54] B. Pes, "Feature selection for high-dimensional data: The issue of stability," *26th IEEE International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises, WETICE*, pp. 170–175, 2017.

[55] U. M. Khaire and R. Dhanalakshmi, "Stability of feature selection algorithm: A review," *Journal of King Saud University - Computer and Information Sciences*, 2019.

[56] "Cosar activity recognition dataset." [Online]. Available: https://everywarelab.di.unimi.it/index.php/palspot

[57] "Human activity recognition using smartphones dataset." [Online]. Available: https://archive.ics.uci.edu/ml/datasets/Human+Activity+Recognition+Using+Smartphones

[58] "Smartphone dataset for human activity recognition in ambient assisted living." [Online]. Available: https://archive.ics.uci.edu/ml/datasets/Smartphone+Dataset+for+Human+Activity+Recognition+(HAR)+in+Ambient+Assisted+Living+(AAL)

[59] "Smartphone-based recognition of human activities and postural transitions dataset." [Online]. Available: http://archive.ics.uci.edu/ml/datasets/smartphone-based+recognition+of+human+activities+and+postural+transitions

[60] "Daily and sports activities dataset." [Online]. Available: http://archive.ics.uci.edu/ml/datasets/Daily+and+Sports+Activities

[61] V. Bolón-Canedo, N. Sánchez-Maroño, and A. Alonso-Betanzos, *Feature Selection for High-Dimensional Data (Artificial Intelligence: Foundations, Theory, and Algorithms)*. Cham, Switzerland: Springer, 2015.

[62] B. Pes, "Ensemble feature selection for high-dimensional data: a stability analysis across multiple domains," *Neural Computing and Applications*, vol. 32, pp. 5951–5973, 2020.

[63] L. Jatoba, U. Grossmann, C. Kunze, J. Ottenbacher, and W. Stork, "Context-aware mobile health monitoring: Evaluation of different pattern recognition methods for classification of physical activity," *30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pp. 5250–5253, 2008.

[64] U. Maurer, A. Smailagic, D. Siewiorek, and M. Deisher, "Activity recognition and monitoring using multiple sensors on different body positions," in *Proc. International Workshop on Wearable and Implantable Body Sensor Networks*, 2006, pp. 113–116.

[65] S. Fan, Y. Jia, and C. Jia, "A feature selection and classification method for activity recognition based on an inertial sensing unit," *Information*, vol. 10, no. 10, p. 290, 2019.

[66] H. Nweke, Y. Teh, and G. Mujtaba, "Multi-sensor fusion based on multiple classifier systems for human activity identification," *Hum. Cent. Comput. Inf. Sci*, vol. 9, p. 34, 2019.

[67] M. Zhang and A. A. Sawchuk, "A feature selection-based framework for human activity recognition using wearable multimodal sensors," in *Proceedings of the 6th International Conference on Body Area Networks*. Brussels, BEL, p: ser. BodyNets '11, 2011, pp. 92–98.

[68] J. Tang, S. Alelyani, and H. Liu, "Feature selection for classification: A review," in *Data Classification:Algorithms and Applications*, C. C. Aggarwal, Ed. CRC Press, 2014, pp. 37–64.

[69] N. Dessì and B. Pes, "Similarity of feature selection methods: An empirical study across data intensive classification tasks," *Expert Syst. Appl.*, vol. 42, no. 10, pp. 4632–4642, 2015.

[70] J. Li, K. Cheng, S. Wang, F. Morstatter, R. P. Trevino, J. Tang, and H. Liu, "Feature selection: A data perspective," *ACM Computing Surveys*, vol. 50, no. 6, pp. 1–45, 2018.

[71] V. Bolón-Canedo and A. Alonso-Betanzos, "Ensembles for feature selection: A review and future trends," *Information Fusion*, vol. 52, pp. 1–12, 2019.

[72] N. Almugren and H. Alshamlan, "A survey on hybrid feature selection methods in microarray gene expression data for cancer classification," *IEEE Access*, vol. 7, pp. 78 533–78 548, 2019.

[73] D. Oreski, S. Oreski, and B. Klicek, "Effects of dataset characteristics on the performance of feature selection techniques," *Applied Soft Computing*, vol. 52, pp. 109–119, 2017.

[74] A. Kalousis, J. Prados, and M. Hilario, "Stability of feature selection algorithms: a study on high dimensional spaces," *Knowl. Inf. Syst.*, vol. 12, no. 1, pp. 95–116, 2007.

[75] W. Awada, T. Khoshgoftaar, D. Dittman, R. Wald, and A. Napolitano, "A review of the stability of feature selection techniques for bioinformatics data," in *IEEE 13th International Conference on Information Reuse and Integration*, 2012, p. 356–363.

[76] L. Kuncheva, "A stability index for feature selection," in *Proceedings of the 25th IASTED International Multi-Conference: Artificial Intelligence and Applications*, A. Press, Ed., 2007, pp. 390–395.

[77] B. Pes and G. Lai, "Cost-sensitive learning strategies for high-dimensional and imbalanced data: a comparative study," *PeerJ Computer Science*, vol. 7, p. e832, 2021.

[78] P.-N. Tan, M. Steinbach, A. Karpatne, and V. Kumar, *Introduction to data mining*, 2nd ed. Pearson, 2019.

[79] I. H. Witten, E. Frank, M. A. Hall, and C. J. Pal, *Data Mining: practical machine learning tools and techniques*. Morgan Kaufmann, 2016.

[80] R. C. Holte, "Very simple classification rules perform well on most commonly used datasets," *Mach. Learn*, vol. 11, pp. 63–91, 1993.

[81] R. J. Urbanowic, M. Meeker, W. L. Cava, R. S. Olson, and J. H. Moore, "Relief-based feature selection: Introduction and review," *Journal of Biomedical Informatics*, vol. 85, pp. 189–203, 2018.

[82] A. Rakotomamonjy, "Variable selection using svm based criteria," *Journal of Machine Learning Research*, vol. 3, pp. 1357–1370, 2003.

[83] V. Bolón-Canedo, N. Sánchez-Maroño, and A. Alonso-Betanzos, "Recent advances and emerging challenges of feature selection in the context of big data," *Knowledge-based systems*, vol. 86, pp. 33–45, 2015.

[84] I. Guyon, S. Gunn, M. Nikravesh, and L. A. Zadeh, *Feature extraction: foundations and applications*. Springer, 2008, vol. 207.

[85] "Weka 3: Data mining software in java. [online]. available:." [Online]. Available: https://www.cs.waikato.ac.nz/ml/weka/

[86] S. S. Keerthi, S. K. Shevade, C. Bhattacharyya, and K. R. K. Murthy, "Improvements to platt's smo algorithm for svm classifier design," *Neural Computation*, vol. 13, no. 3, pp. 637–649, 2001.

[87] B. Pes, "Learning from high-dimensional and class-imbalanced datasets using random forests," *Information*, vol. 12, no. 8, p. 286, 2021.

[88] D. Anguita, A. Ghio, L. Oneto, X. Parra, and J. Reyes-Ortiz, "A public domain dataset for human activity recognition using smartphones," in *Proceedings of the 21th International European Symposium on Artificial Neural Networks*, C. Intelligence and M. Learning, Eds., 2013, pp. 437–442.

[89] K. Altun, B. Barshan, and O. Tunçel, "Comparative study on classifying human activities with miniature inertial and magnetic sensors," *Pattern Recognition*, vol. 43, no. 10, pp. 3605–3620, October 2010.

[90] B. Barshan and M. C. Yüksek, "Recognizing daily and sports activities in two open source machine learning environments using body-worn sensor units," *The Computer Journal*, vol. 57, no. 11, pp. 1649–1667, November 2014.

[91] S. Nogueira, K. Sechidis, and G. Brown, "On the stability of feature selection algorithms." *J. Mach. Learn. Res.*, vol. 18, no. 1, pp. 6345–6398, 2017.

[92] L. Chen, S. Fan, V. Kumar, and Y. Jia, "A method of human activity recognition in transitional period," *Information*, vol. 11, no. 9, p. 416, 2020.

[93] Y. Tian, J. Zhang, L. Li, and Z. Liu, "A novel sensor-based human activity recognition method based on hybrid feature selection and combinational optimization," *IEEE Access*, vol. 9, pp. 107 235–107 249, 2021.

[94] E. Mocanu, P. H. Nguyen, M. Gibescu, and W. L. Kling, "Comparison of machine learning methods for estimating energy consumption in buildings," in *2014 International Conference on Probabilistic Methods Applied to Power Systems (PMAPS)*, 2014, pp. 1–6.

[95] D. Bhatt, D. D, A. Hariharasudan, M. Lis, and M. Grabowska, "Forecasting of energy demands for smart home applications," *Energies*, vol. 14, no. 4, 2021. [Online]. Available: https://www.mdpi.com/1996-1073/14/4/1045

[96] V. Dehalwar, A. Kalam, M. L. Kolhe, and A. Zayegh, "Electricity load forecasting for urban area using weather forecast information," in *2016 IEEE International Conference on Power and Renewable Energy (ICPRE)*. IEEE, 2016, pp. 355–359.

[97] H. Ziekow, C. Goebel, J. Strüker, and H.-A. Jacobsen, "The potential of smart home sensors in forecasting household electricity demand," in *2013 IEEE International Conference on Smart Grid Communications (SmartGridComm)*, 2013, pp. 229–234.

[98] A. Barbato, A. Capone, M. Rodolfi, and D. Tagliaferri, "Forecasting the usage of household appliances through power meter sensors for demand management in the smart grid."

[99] R. Dobbe, D. Arnold, S. Liu, D. Callaway, and C. Tomlin, "Real-time distribution grid state estimation with limited sensors and load forecasting," in *2016 ACM/IEEE 7th International Conference on Cyber-Physical Systems (ICCPS)*, 2016, pp. 1–10.

[100] N. C. Truong, J. McInerney, L. Tran-Thanh, E. Costanza, and S. Ramchurn, "Forecasting multi-appliance usage for smart home energy management," in *Twenty-Third International Joint Conference on Artificial Intelligence*, 2013.

[101] J. Wang, Y. Chen, S. Hao, X. Peng, and L. Hu, "Deep learning for sensor-based activity recognition: A survey," *Pattern Recognition Letters*, vol. 119, pp. 3–11, 2019.

[102] A. Keshavarzian, S. Sharifian, and S. Seyedin, "Modified deep residual network architecture deployed on serverless framework of iot platform based on human activity recognition application," *Future Gener. Comput. Syst.*, vol. 101, pp. 14–28, 2019.

[103] M. Rawashdeh, M. G. H. al Zamil, S. Samarah, M. S. Hossain, and G. Muhammad, "A knowledge-driven approach for activity recognition in smart homes based on activity profiling," *Future Gener. Comput. Syst.*, vol. 107, pp. 924–941, 2020.

[104] A. Matassa and D. Riboni, "Reasoning with smart objects' affordance for personalized behavior monitoring in pervasive information systems," *Knowl. Inf. Syst.*, vol. 62, no. 4, pp. 1255–1278, 2020.

[105] G. Civitarese, C. Bettini, T. Sztyler, D. Riboni, and H. Stuckenschmidt, "*newNECTAR*: Collaborative active learning for knowledge-based probabilistic activity recognition," *Pervasive and Mobile Computing*, vol. 56, pp. 88–105, 2019.

[106] D. Riboni and M. Murtas, "Sensor-based activity recognition: One picture is worth a thousand words," *Future Gener. Comput. Syst.*, vol. 101, pp. 709–722, 2019.

[107] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.

[108] S. le Cessie and J. van Houwelingen, "Ridge estimators in logistic regression," *Applied Statistics*, vol. 41, no. 1, pp. 191–201, 1992.

[109] D. J. Cook, A. S. Crandall, B. L. Thomas, and N. C. Krishnan, "CASAS: A smart home in a box," *Computer*, vol. 46, no. 7, pp. 62–69, 2013.

[110] J. Lee Rodgers and W. A. Nicewander, "Thirteen ways to look at the correlation coefficient," *The American Statistician*, vol. 42, no. 1, pp. 59–66, 1988.

[111] G.-F. Angelis, C. Timplalexis, S. Krinidis, D. Ioannidis, and D. Tzovaras, "Nilm applications: Literature review of learning approaches, recent developments and challenges," *Energy and Buildings*, p. 111951, 2022.

[112] A.-R. Al-Ali, I. A. Zualkernan, M. Rashid, R. Gupta, and M. Alikarar, "A smart home energy management system using iot and big data analytics approach," *IEEE Transactions on Consumer Electronics*, vol. 63, no. 4, pp. 426–434, 2017.

[113] D.-M. Han and J.-H. Lim, "Smart home energy management system using ieee 802.15. 4 and zigbee," *IEEE Transactions on Consumer Electronics*, vol. 56, no. 3, pp. 1403–1410, 2010.

[114] F. M. Wittmann, J. C. López, and M. J. Rider, "Nonintrusive load monitoring algorithm using mixed-integer linear programming," *IEEE Transactions on Consumer Electronics*, vol. 64, no. 2, pp. 180–187, 2018.

[115] K. C. Armel, A. Gupta, G. Shrimali, and A. Albert, "Is disaggregation the holy grail of energy efficiency? the case of electricity," *Energy Policy*, vol. 52, pp. 213–234, 2013.

[116] E. Elhamifar and S. Sastry, "Energy disaggregation via learning powerlets and sparse coding." in *AAAI*. AAAI, 2015, pp. 629–635.

[117] S. Gupta, M. S. Reynolds, and S. N. Patel, "Electrisense: single-point sensing using emi for electrical event detection and classification in the home," in *Proceedings of the 12th ACM international conference on Ubiquitous computing*, ACM. ACM, 2010, pp. 139–148.

[118] G. Kalogridis, C. Efthymiou, S. Z. Denic, T. A. Lewis, and R. Cepeda, "Privacy for smart meters: Towards undetectable appliance load signatures," in *Smart Grid Communications (SmartGridComm), 2010 First IEEE International Conference on*, IEEE. IEEE, 2010, pp. 232–237.

[119] A. Prudenzi, "A neuron nets based procedure for identifying domestic appliances pattern-of-use from energy recordings at meter panel," in *Power Engineering Society Winter Meeting, 2002. IEEE*, vol. 2, IEEE. IEEE, 2002, pp. 941–946.

[120] K. Basu, V. Debusschere, A. Douzal-Chouakria, and S. Bacha, "Time series distance-based methods for non-intrusive load monitoring in residential buildings," *Energy and Buildings*, vol. 96, pp. 109–117, 2015.

[121] K. Basu, V. Debusschere, S. Bacha, U. Maulik, and S. Bondyopadhyay, "Non-intrusive load monitoring: A temporal multilabel classification approach," *IEEE Transactions on Industrial informatics*, vol. 11, no. 1, pp. 262–270, 2015.

[122] H. Kim, M. Marwah, M. Arlitt, G. Lyon, and J. Han, "Unsupervised disaggregation of low frequency power measurements," in *Proceedings of the 2011 SIAM international conference on data mining*, SIAM. SIAM, 2011, pp. 747–758.

[123] O. Parson, S. Ghosh, M. Weal, and A. Rogers, "Non-intrusive load monitoring using prior models of general appliance types," in *Twenty-Sixth AAAI Conference on Artificial Intelligence*. AAAI, 2012.

[124] A. Cominola, M. Giuliani, D. Piga, A. Castelletti, and A. E. Rizzoli, "A hybrid signature-based iterative disaggregation algorithm for non-intrusive load monitoring," *Applied energy*, vol. 185, pp. 331–344, 2017.

[125] L. Massidda and M. Marrocu, "A bayesian approach to unsupervised, non-intrusive load disaggregation," *Sensors*, vol. 22, no. 12, p. 4481, 2022.

[126] F. Hidiyanto and A. Halim, "Knn methods with varied k, distance and training data to disaggregate nilm with similar load characteristic," in *Proceedings of the 3rd Asia Pacific Conference on Research in Industrial and Systems Engineering 2020*. ACM, 2020, pp. 93–99.

[127] M. Singh, S. Kumar, S. Semwal, and R. Prasad, "Residential load signature analysis for their segregation using wavelet—svm," in *Power Electronics and Renewable Energy Systems*. Springer, 2015, pp. 863–871.

[128] F. Gong, N. Han, Y. Zhou, S. Chen, D. Li, and S. Tian, "A svm optimized by particle swarm optimization approach to load disaggregation in non-intrusive load monitoring in smart homes," in *2019 IEEE 3rd Conference on Energy Internet and Energy System Integration (EI2)*, IEEE. IEEE, 2019, pp. 1793–1797.

[129] M. Hasan, D. Chowdhury, M. Khan, Z. Rahman *et al.*, "Non-intrusive load monitoring using current shapelets," *Applied Sciences*, vol. 9, no. 24, p. 5363, 2019.

[130] Z. Xiao, W. Gang, J. Yuan, Y. Zhang, and C. Fan, "Cooling load disaggregation using a nilm method based on random forest for smart buildings," *Sustainable Cities and Society*, vol. 74, p. 103202, 2021.

[131] X. Wu, Y. Gao, and D. Jiao, "Multi-label classification based on random forest algorithm for non-intrusive load monitoring system," *Processes*, vol. 7, no. 6, p. 337, 2019.

[132] D. Murray, L. Stankovic, V. Stankovic, S. Lulic, and S. Sladojevic, "Transferability of neural network approaches for low-rate energy disaggregation," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE. IEEE, 2019, pp. 8330–8334.

[133] M. D'Incecco, S. Squartini, and M. Zhong, "Transfer learning for non-intrusive load monitoring," *IEEE Transactions on Smart Grid*, vol. 11, no. 2, pp. 1419–1429, 2019.

[134] J. Jiang, Q. Kong, M. D. Plumbley, N. Gilbert, M. Hoogendoorn, and D. M. Roijers, "Deep learning-based energy disaggregation and on/off detection of household appliances," *ACM Transactions on Knowledge Discovery from Data (TKDD)*, vol. 15, no. 3, pp. 1–21, 2021.

[135] J. Song, H. Wang, M. Du, L. Peng, S. Zhang, and G. Xu, "Non-intrusive load identification method based on improved long short term memory network," *Energies*, vol. 14, no. 3, p. 684, 2021.

[136] H. Çimen, N. Çetinkaya, J. C. Vasquez, and J. M. Guerrero, "A microgrid energy management system based on non-intrusive load monitoring via multitask learning," *IEEE Transactions on Smart Grid*, vol. 12, no. 2, pp. 977–987, 2020.

[137] M. Valenti, R. Bonfigli, E. Principi, and S. Squartini, "Exploiting the reactive power in deep neural models for non-intrusive load monitoring," in *2018 International Joint Conference on Neural Networks (IJCNN)*, IEEE. IEEE, 2018, pp. 1–8.

[138] A. Faustine, L. Pereira, H. Bousbiat, and S. Kulkarni, "Unet-nilm: A deep neural network for multi-tasks appliances state detection and power estimation

in nilm," in *Proceedings of the 5th International Workshop on Non-Intrusive Load Monitoring*. ACM, 2020, pp. 84–88.

[139] Z. Yue, C. R. Witzig, D. Jorde, and H.-A. Jacobsen, "Bert4nilm: A bidirectional transformer model for non-intrusive load monitoring," in *Proceedings of the 5th International Workshop on Non-Intrusive Load Monitoring*. ACM, 2020, pp. 89–93.

[140] V. Piccialli and A. M. Sudoso, "Improving non-intrusive load disaggregation through an attention-based deep neural network," *Energies*, vol. 14, no. 4, p. 847, 2021.

[141] L. Massidda, M. Marrocu, and S. Manca, "Non-intrusive load disaggregation by convolutional neural network and multilabel classification," *Applied Sciences*, vol. 10, no. 4, p. 1454, 2020.

[142] ——, "Non-intrusive load disaggregation via a fully convolutional neural network: improving the accuracy on unseen household," in *2020 2nd IEEE International Conference on Industrial Electronics for Sustainable Energy Systems (IESES)*, vol. 1, IEEE. IEEE, 2020, pp. 317–322.

[143] R. Terracciano, V. Galdi, V. Calderaro, D. Pappalardo, G. Ceneri, and A. O. Pití, "Demand side management services for smart buildings with the use of second generation smart meter and the chain-2 of e-distribuzione," in *2020 IEEE International Conference on Environment and Electrical Engineering and 2020 IEEE Industrial and Commercial Power Systems Europe (EEEIC / I CPS Europe)*. IEEE, 2020, pp. 1–6.

[144] D. Serra, D. Mardero, L. Di Stefano, and S. Grillo, "Post-metering value-added services for low voltage electricity users: Lessons learned from the italian experience of chain 2," *Applied Energy*, vol. 304, p. 117806, Dec 2021. [Online]. Available: http://dx.doi.org/10.1016/j.apenergy.2021.117806

[145] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*. IEEE, 2017, pp. 2881–2890.

[146] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "Pytorch: An imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems 32*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, Eds. Curran Associates, Inc., 2019, pp. 8024–8035. [Online]. Available: http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf

[147] S. Vitiello, N. Andreadou, M. Ardelean, and G. Fulli, "Smart metering roll-out in europe: Where do we stand? cost benefit analyses in the clean energy package and research trends in the green deal," *Energies*, vol. 15, no. 7, p. 2340, 2022.

[148] C. Staff, "Cei-en 50065-1, signalling on low-voltage electrical installations in the frequency range 3 khz to 148,5 khz part 1: General requirements, frequency bands and electromagnetic disturbances," *CEI Standards*, 2012.

[149] ——, "Cei ts 13-82:2017-08, sistemi di misura dell'energia elettrica - comunicazione con i dispositivi utente, parte 2: Modello dati e modello applicativo." *CEI Standards*, 2012.

[150] J. Kelly and W. Knottenbelt, "The uk-dale dataset, domestic appliance-level electricity demand and whole-house demand from five uk homes," *Scientific data*, vol. 2, p. 150007, 2015.

[151] D. Murray, L. Stankovic, and V. Stankovic, "An electrical load measurements dataset of united kingdom households from a two-year longitudinal study," *Scientific data*, vol. 4, no. 1, pp. 1–12, 2017.

[152] P. Laviron, X. Dai, B. Huquet, and T. Palpanas, "Electricity demand activation extraction: From known to unknown signatures, using similarity search," in *Proceedings of the Twelfth ACM International Conference on Future Energy Systems*. ACM, 2021, pp. 148–159.

[153] H. Rafiq, X. Shi, H. Zhang, H. Li, and M. K. Ochani, "A deep recurrent neural network for non-intrusive load monitoring based on multi-feature input space and post-processing," *Energies*, vol. 13, no. 9, p. 2195, 2020.

[154] G. Zhou, Z. Li, M. Fu, Y. Feng, X. Wang, and C. Huang, "Sequence-to-sequence load disaggregation using multiscale residual neural network," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–10, 2020.

[155] C. Puente, R. Palacios, Y. González-Arechavala, and E. F. Sánchez-Úbeda, "Non-intrusive load monitoring (nilm) for energy disaggregation using soft computing techniques," *Energies*, vol. 13, no. 12, p. 3117, 2020.

[156] Y. Pan, K. Liu, Z. Shen, X. Cai, and Z. Jia, "Sequence-to-subsequence learning with conditional gan for power disaggregation," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE. IEEE, 2020, pp. 3202–3206.

[157] F. Marchesoni-Acland, C. Mariño, E. Masquil, P. Masaferro, and A. Fernández, "End-to-end nilm system using high frequency data and neural networks," 2020. [Online]. Available: https://arxiv.org/abs/2004.13905

[158] X. Wu, D. Jiao, K. Liang, and X. Han, "A fast online load identification algorithm based on vi characteristics of high-frequency data under user operational constraints," *Energy*, vol. 188, p. 116012, 2019.

[159] R. Medico, L. De Baets, J. Gao, S. Giri, E. Kara, T. Dhaene, C. Develder, M. Bergés, and D. Deschrijver, "A voltage and current measurement dataset for plug load appliance identification in households," *Scientific Data*, vol. 7, no. 1, pp. 1–10, Feb. 2020.

[160] A. Faustine and L. Pereira, "Multi-label learning for appliance recognition in nilm using fryze-current decomposition and convolutional neural network," *Energies*, vol. 13, no. 16, p. 4154, 2020.

[161] D. F. Teshome, T. D. Huang, and K.-L. Lian, "Distinctive load feature extraction based on fryze's time-domain power theory," *IEEE Power and Energy Technology Systems Journal*, vol. 3, no. 2, pp. 60–70, 2016.

[162] P. Mehta, M. Bukov, C.-H. Wang, A. G. Day, C. Richardson, C. K. Fisher, and D. J. Schwab, "A high-bias, low-variance introduction to machine learning for physicists," *Physics reports*, vol. 810, p. 32, 2019.

[163] L. Pereira and N. Nunes, "A comparison of performance metrics for event classification in Non-Intrusive Load Monitoring," in *2017 IEEE International Conference on Smart Grid Communications (SmartGridComm)*, Oct. 2017, pp. 159–164.

[164] ——, "An empirical exploration of performance metrics for event detection algorithms in Non-Intrusive Load Monitoring," *Sustainable Cities and Society*, vol. 62, p. 102399, Nov. 2020.