UNIVERSITY OF L'AQUILA

DEPARTMENT OF INDUSTRIAL AND INFORMATION ENGINEERING AND ECONOMICS

Doctor of philosophy course on Industrial and Information Engineering and Economics:
Curriculum Ingegneria elettrica, elettronica e dell'informazione
XXXV cycle

Innovative Machine Learning Systems for Non-Intrusive Load Monitoring

SSD ING-INF/07

Ph. D Student:
Simone Mari

_____

Coordinator of the course
Prof. Katia Gallucci

_____

Tutor
Prof. Giovanni Bucci

_____

A.A. 2021/2022

Abstract

Non-Intrusive Load Monitoring (NILM) is the process that allows obtaining information about the electrical loads powered by an electrical system through a single measurement performed in a single point of the system itself.

Systems based on this process provide an alternative solution to the more traditional intrusive one. NILM requires a reduced number of equipment and less occupied space, even if it presents a greater complexity in terms of processing the acquired data. In fact, this solution is much simpler, from the hardware point of view, as it requires the measurement of a voltage and a current, or often even just the current. However, the complexity shifts to the processing section, which must identify the absorption of the individual devices through the use of appropriate algorithms.

The information required from an electrical loads monitoring system may concern their status (ON/OFF) or the electrical quantities involved in their operation. This information must be made available in a more or less short time depending on the application in which the measuring system is used.

The most common application is to monitor the electricity consumption of different devices within a residential home. In this case the information must be updated on time intervals of the order of days or weeks.

Today, new NILM systems are used in numerous innovative applications in residential environments. For example, some human activity recognition (HAR) and ambient assisted living (AAL) systems are based on disaggregated appliance activity data. Innovative commercial and industrial applications are based on the NILM technique, such as to implement predictive maintenance. Energy disaggregation is also applied to manage the generation and storage of energy in smart grids. Therefore, the times in which it is necessary to have information about the state or the electrical quantities of a load are drastically reduced, down to a few seconds.

In the first part of this thesis the current state-of-the-art of NILM systems will be defined, paying particular attention to the most significant contributions. Subsequently, the applications of these systems in industrial and residential contexts will be described in detail.

In the second part, three different systems will be proposed having different characteristics both from the point of view of the electrical quantities measured, the sampling frequency and the signal processing section. More specifically, the experimental systems created, based on a microcontroller, use Machine Learning algorithms to process the signals obtained from the measurement section. For each of the proposed systems, a wide range of measurements on test systems were carried out, in order to effectively evaluate their performance in real conditions.

In the references, for better clarity, the publications resulting from this thesis are reported at the beginning [1]-[12].

Table of Content

Chapter 1

Introduction

The analysis of the energy consumed by individual devices powered by an electrical system makes it possible to identify the least efficient or malfunctioning devices, and to implement the appropriate actions aimed at reducing electrical energy consumption. Furthermore, it is possible to identify the maximum absorption peaks and their allocation over time.

An analysis of this type can be conducted through intrusive load monitoring (ILM) systems or non-intrusive load monitoring (NILM) systems.

In the first systems, a transducer is installed on each equipment, to measure the energy consumption of each individual load. These systems provide very accurate results, but are often too complicated to implement in an existing system, mainly due to space constraints that do not allow easy installation of transduction and communication systems.

Non-intrusive systems measure the total absorbed power, from which the individual contributions relating to each load are “disaggregated”. In other words, the specific energy consumption model of the different electrical loads, referred to as "signature", is utilized. This second solution is much simpler, from the hardware point of view, as it requires the measurement of a voltage and a current, or often even just the current. However, the complexity shifts to the processing section, which must identify the absorptions of the individual devices through the use of suitable algorithms [6].

The type of analysis to be applied also depends on the type of installation, which must be defined first. The NILM system can be placed in the switch box inside a property or even at a long distance from it. In the first case, the NILM is a miniaturized system with an embedded microcontroller. This system is capable of acquiring and processing the signals locally. In the second case, the system is a computer, which processes the data transmitted to a cloud database by the local

smart energy meter. Local systems are capable of acquiring voltage and current (even with sampling frequencies of some kilohertz) processing them in real-time and displaying the results or storing them on a remote server [13]. Remote systems can only use data (normally related only to the active power) available in the cloud with measurement frequencies ranging from 1 Hz to 3 Hz due to limited data transmission and storage capabilities [14].

It is important to acknowledge that the widespread availability of NILM measurement systems is currently a significant challenge in the industry. Despite the presence of companies developing NILM solutions [15]-[17], they tend to focus on business-to-business (B2B) services, rather than offering business-to-consumer (B2C) sales of hardware. This is primarily due to the fact that NILM technology is mainly utilized for energy management and monitoring in commercial and industrial settings, rather than in residential homes. These companies typically offer a wide range of services, such as energy audits, monitoring and reporting, and energy efficiency consulting, to businesses and organizations. This approach allows them to closely collaborate with customers to gain a deep understanding of their specific energy usage patterns, and provide tailored solutions aimed at reducing energy consumption and costs. However, this presents a problem as it makes it difficult to compare the NILM systems proposed by research with commercially available NILM systems, as the latter are not readily accessible to the general public.

The International Electrotechnical Commission (IEC) has defined a standard [18] to provide classification of NILM sensing devices for use in NILM systems, according to the state of the art of NILM technologies. As defined by the standard, a NILM system consists of:

- a NILM sensing device (NSD) connected to the electrical installation and producing data relevant for load signature identification;
- NILM analytics using the data output from the NSD and producing information to the users about their energy usage.

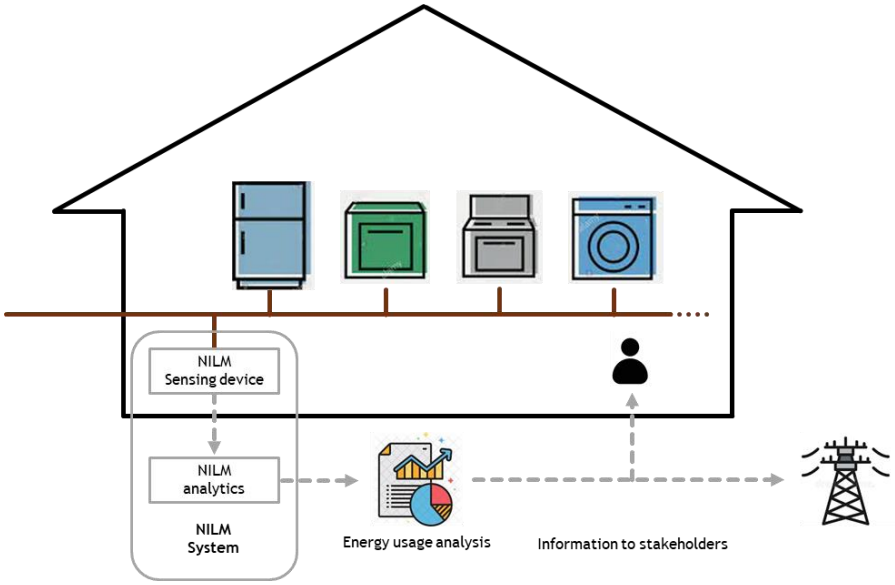

Figure 1.1: Element of a NILM System.

The performance of the NILM system depends on the characteristics of the NSD and on the characteristics of the NILM analytics.

The standard classifies NSD according to three essential parameters:

- *input sampling frequency*, the frequency at which the electrical signals are sampled by the NSD;
- *output rate*, the rate at which the NSD produces data that can be used by NILM analytics;
- *data bit rate*, the average bit-per-second (bps) over an hour at which the electrical signals are quantified by the NSD.

It also defines the classification of NSDs based on the three essential parameters according to Table 1.1, Table 1.2 and Table 1.3.

Table 1.1: Classification of NSDs according to the input sampling frequency.

Input sampling frequency f	**Parameters only**	**< 5 kHz**	**5 kHz ≤ f < 8 kHz**	**8 kHz ≤ f < 100 kHz**	**100 kHz ≤ f < 1 MHz**	**≥ 1 MHz**
Class	P	1	2	3	4	5

Table 1.2: Classification of NSDs according to output data rate.

Output data rate *d*	**> 30 min**	**30 min≧ *d* > 1 min**	**1 min ≧ *d* > 1 s**	**1 s ≧ *d* > 0.1 s**	**≤ 0.1 s**
Class	E	D	C	B	A

Table 1.3: Classification of NSDs according to the data bit rate.

Data bit rate *b*	**< 100 bps**	**100 bps ≤ *b* < 1 kbps**	**1 kbps ≤ *b* < 10 kbps**	**10 kbps ≤ *b* < 100 kbps**	**≥ 100 kbps**
Class	L	M	H	S	X

NSDs belonging to class P do not provide samples of electrical waveforms, but rather provide measurements or estimates of electrical parameters (such as active power, reactive power, power factor, harmonic distortion, etc.). An example of NSDs belonging to class P are classic power meters for residential use that provide only measurements of electrical parameters. On the other hand, power quality instrumentation or dedicated measurement systems can also provide waveforms in detail. Therefore, in this case they would fall into classes 1, 2, 3, 4 or 5 depending on their sampling rate.

According to the standard, if an NSD includes the ability to work at multiple sample rates, output data rates or data bit rates, the manufacturer must provide a class for each.

It also specifies that for NSDs using information related to transient events, the data bit rate depends on the occurrence of the transient events, and the rate of occurrence used for classification shall be indicated.

The core of NILM systems lies in the analytics section, thus in its ability to use the data produced by the NSD.

The NILM problem can be formulated as in (1.1).

$$Y(t) = \sum_{n=1}^{M} y_n(t) + \varepsilon(t) \tag{1.1}$$

where $Y(t)$ is the measured aggregate signal, $y_n(t)$ is the contribution of the n-th of the M loads, and $\varepsilon(t)$ is the sum of the noise and estimation error. The purpose of a NILM system is to obtain $y_n(t)$ signals from measuring only the $Y(t)$ signal.

Electric loads have a unique energy consumption scheme, or signature, as shown in Fig.1.2, which allows the disaggregation algorithms to distinguish them starting from the aggregate load measurements. The effectiveness of the disaggregation algorithm improves with the uniqueness of the signatures.

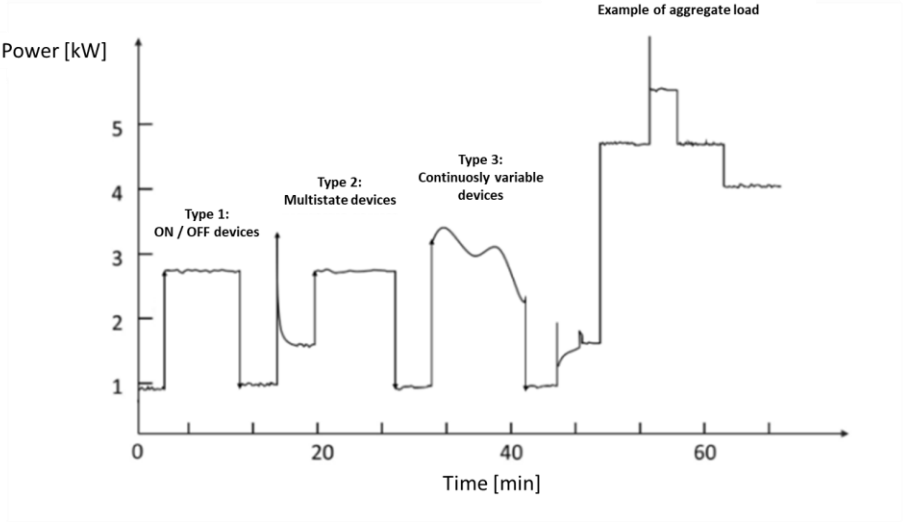

Figure 1.2: Aggregate load due to different individual load types.

The concept of NILM, introduced by G. W. Hart in the early 1980s, has been the subject of research for several years. Since 2010 there has been a great focus on these research topics, mainly oriented to the residential sector. Despite the growing importance of energy efficiency, the residential sector still lags behind in adopting load monitoring systems. This leads to a pervasive use of energy-inefficient equipment, thus limiting the potential for energy savings and increased efficiency. As proposed by Hart [19], appliances can be classified according to their operational states as follows:

Type 1) electric loads with two operating states (ON / OFF);

Type 2) electric loads with a finite number of operating states, also called finite state machines (FSM). The switching pattern of these appliances makes it possible for the disaggregation algorithm to identify their operation;

Type 3) electrical loads with continuously variable power absorption, without a fixed number of states, therefore known as continuously variable devices (CVD). Therefore, it is very difficult to disaggregate this type of equipment from aggregate load measurements;

Type 4) electric loads that remain active for days or weeks, consuming energy at a constant rate, and are therefore called “permanent consumption devices” [20].

The use of traditional algorithms makes it easy to recognize Type 1 electrical loads, as was also proposed by Hart in his early work. Difficulties arise when Type 2 and particularly Type 3 electrical loads are present, which therefore require appropriate modeling. Over the past decade, advances in parallel processing and machine learning (ML), as well as greater availability of computing resources, have enabled the development of increasingly sophisticated algorithms for estimating energy consumption. These systems have been able to process large amounts of data, using artificial intelligence algorithms to identify consumption patterns and estimate the behaviors of each electrical loads. This made it possible to develop NILM systems capable of estimating the consumption of different types of electrical loads with a higher level of accuracy than traditional algorithms.

Chapter 2 will address in detail the current state of the art of algorithms used for energy disaggregation.

Chapter 2

State-of-the-art in methods for energy disaggregation

This section analyzes the state of the art relating to NILM systems. Various techniques have been proposed in the literature, which differ in the sampling frequency of the signals, in the different approach used to recognize the device, in the algorithmic technique used. The proposed techniques have evolved over time; it is therefore interesting to analyze the state of the art following this evolution.

Since Hart proposed the first NILM system in the 1980s, there have been significant developments. Over the next two decades, research on this topic focused on finding new signatures capable of uniquely identifying devices (active power, reactive power, power factor, harmonic content, inactive currents, transient information, etc.) for then develop classifiers capable of providing indications on the basis of these signatures. This type of approach often involves the detection of events upstream of their classification.

Once an event has been detected, the features (and then the signature) associated with the appliance that caused it are extracted. This type of approach can therefore be divided into three basic steps: event detection, feature extraction and load identification.

Consequently, these techniques can be grouped within a framework called "event-based". By "event" is meant a change of electrical parameters in the aggregate signal. This definition was needed when, from 2010 onwards, researchers started proposing "non-event based" approaches. These systems therefore no longer included the event detection phase. In these systems the concept of "features / signature" is lost, as the features used by the models are the aggregate power signal itself.

Deep Learning (DL), Factorial Hidden Markov Models (FHMM) and Discriminative Sparse Coding (DSC) based systems are examples of algorithm groups that fall into the "non-event based" category.

Over the years, manuscripts that have proposed overviews of NILM systems have used different criteria for the subdivision of approaches.

In addition to this subdivision linked to the "event based" or "non-event based" approaches, another important approach refers to the sampling frequency, which defines "high frequency" and "low frequency" techniques.

In general, "low frequency" refers to sampling rates of 1 Hz or less. Almost all "non-event based" systems fall into this category, conversely almost all "event-based" systems fall into the "high frequency" category. However, there are "event-based" systems that involve processing the sampled signal in low frequency. The sampling frequency of 1 Hz guarantees a good resolution as regards the detection of the transition.

The "high frequency" techniques have a sampling frequency that exceeds 100/120 Hz. Consequently, there is an almost total absence of proposed algorithms operating in the frequency between 1 Hz and 100/120 Hz. Sampling above 100/120 Hz makes it possible to extract information on the harmonic content of voltage and current, useful for classifying some types of devices. With higher sampling rates (of the order of ten kilohertz) it is also possible to analyze transient characteristics on electrical signals such as overshoot, rise times, settling times, etc.

A further subdivision, often used in literature, is that of NILM systems based on "macroscopic" and "microscopic" characteristics, referring to the extractable characteristics from the measured signals at "low frequency" and "high frequency" respectively.

2.1 Event-based methods

The Event-based NILM systems provide for the classification of an appliance starting from the detection of an event caused by it. As mentioned above, these systems have three distinct steps: event detection, feature extraction and load identification. To correctly apply this technique, it is necessary to define a mathematical characterization, and therefore a signature, of the appliances to be recognized. These signatures will then be compared, through a classifier, with the one extracted following an event. After the identification of the load, the energy disaggregation takes place through the knowledge of the operating sequences of the appliance.

2.1.1 Event detection

The techniques that propose algorithms for detecting events by means of high sampling frequency measurements have been discussed in an extensive literature. The papers [21] and [22] can be mentioned as a basic reference.

In [21] an event detection system based on goodness-of-fit (GOF) test is proposed. The authors use a sliding window technique on the measured power signal by comparing each window with the previous one. The purpose of the test is to determine the similarity between the two distributions originating from the measured sample windows.

Given the two distributions, the χ^2 test is carried out, by adding the square of the differences between the theoretical and observed frequencies, weighted on the theoretical frequencies:

$$\chi^2 = \sum_{i=1}^{k} \frac{(x_i - y_i)^2}{y_i} > {\chi^2}_{threshold} \tag{2.1}$$

where x_i and y_i are the frequencies relative to the *i*-th bins of the theoretical distribution and of the observed distribution, respectively. ${\chi^2}_{threshold}$ is the threshold beyond which the presence or absence of an event is established.

In [22] the authors propose a NILM system that acquires voltage and current at a sampling frequency of 30 kHz. The DBSCAN clustering algorithm is used to determine the presence of an event by identifying two adjacent steady states. The authors state that this system is insensitive to outliers due to transients and is able to detect small changes in power, as the algorithm does not provide for the use of thresholds. Once the event is detected, the system creates a vector consisting of 50 elements representing: the rms values of both the total and the fundamental currents,

the proportion of the 1st - 45th order of the rms current value with respect to the rms value, the value of the fundamental current and the cosine value of the 1st, 3rd and 5th order of the phase angle of the harmonic current. This vector will constitute the input of an ANN which will identify the load.

In [23] the authors propose an event-based system using a low sampling rate (1/3 Hz). The events are detected by observing the maximum and minimum peaks, also defining the rise and fall intervals in order to compare the power differences detected in the events with an established threshold.

2.1.2 Features extraction

After the detection of an event, the system proceeds with the extraction of some characteristics with which to obtain a signature of the devices. The parameters of interest can be related to both stationary and transitory conditions. Over the years, a significant effort in NILM research has focused on exploring different signatures.

The characteristics in stationary conditions are typically measured around the detected event, adopting a low sampling frequency. Examples of features in steady state conditions are changes in active and reactive power, power factor, changes in rms current or rms voltage. These features have been widely used in literature to build an electric signature in the form of vectors or matrices [6],[19],[24]-[33].

While these features are easy to extract, they often have an overlap problem. The term overlap refers to the condition in which an electrical signature fails to uniquely represent a load. For example, two appliances that absorb the same active power with a high power factor can often lead to misidentification.

Furthermore, the behavior of the appliances in steady state conditions is not well defined, so that numerous ambiguities may arise regarding the electrical power supply parameters and their operating cycle.

It is also demonstrated that the steady-state features are susceptible to power disturbances [34].

The detection of the transient characteristics allows to overcome the problem, as the analysis is performed only at the on and off transients of the devices. These characteristics, however, require particularly high sampling frequencies, in order to correctly detect the trend of the signal during the transitory. The reduced duration of these phenomena means that, in most cases, the simultaneous insertion of different devices can be excluded. Conversely, the measurement of these features requires a more complex hardware configuration and higher computing requirements.

Examples of works in which electrical signatures have been built with transient features are [34]-[45].

Not even the transitory characteristics, however, proved to be fully capable of uniquely representing household appliances. Therefore, approaches based on hybrid signature composed of features in steady state and transient conditions have been proposed [46]-[49].

In addition to the traditional characteristics, transformation techniques are adopted such as Short Time Fourier Transform (STFT) [50], analysis based on the cepstrum [51], Wavelet Transform (WT) [34] and S Transform (ST) [52] on the aggregate signal to get some advanced, unique and hidden features.

A brief description of the different features used in the literature for NILM is described as follows:

1) *Active Power P*: This characteristic is generally integrated by measuring the duration and the appliance's frequency of use.
2) *P–Q Plan*: Step changes in the active and reactive power *Q* allow easy identification of the ON / OFF status of high-power equipment.
3) *Combination of the P–Q Plane with Extended Transient Characteristics*: It is suitable in identifying devices with relatively long transients and significant peaks of power.
4) *Characteristics Based on P, Q, I, and V at Low Frequencies*: These combinations exhibit good performance in identifying ON / OFF appliances.
5) *P–Q and Harmonic Planes*: The harmonic content or the spectrum of high-frequency sampled currents is usually combined with the *P–Q* characteristics.
6) *Short-Time Fourier Transform (STFT)*: The spectral envelopes allow the identification of nonlinear and variable load devices.
7) *V–I trajectories*: It is suitable in identifying loads, starting from the signal shape.
8) *Nonactive current*: It is suitable in identifying some special equipment.
9) *Unconventional features*: The analysis of voltage–noise spectrum or electromagnetic interference voltage noise has been proposed by several authors.

2.1.3 Load identification

The last step in an event-based NILM system is load identification, which is performed in most cases using ML algorithms that perform well as classification systems.

As described in [23], in a ML based appliance identification technique, a classification algorithm identifies the appliance using the signatures obtained from the event. Numerous supervised ML algorithms have been proposed in literature

including, K Nearest Neighbor (KNN) [24]-[26],[31],[44],[53], Naive Bayes [30][31], Decision Tree (DT) [34],[54],[55], Support Vector Machine (SVM) [4],[10],[26][56]-[58], Principal Component Analysis (PCA) [59] and Artificial Neural Network (ANN) [22],[27],[42],[44],[56],[54],[60],[61].

Differential Evolution and PSO have been proposed to adjust the training parameters of the model in order to speed up the training phase of the ANNs and improve its accuracy [32],[56],[61]. Other ML algorithms that have been used as classifiers are random forest [62], CNN [1],[9],[63],[64], Long Short Term Memory (LSTM) networks [65]-[67], self organizing map [49],[68],[69], competitive autoassociative ANN [70][70] and bagging and boosting [56],[71],[72]. In some works, hybrid or multiple classifiers have been proposed with the aim of improving the accuracy in identifying loads as in [47],[73]. Attempts have been made for handling noisy data using fuzzy logic based classifiers as in [45],[74]. A hybrid Neuro-fuzzy classifier with fuzzy C means clustering is reported in [73] to handle appliance classification in ambiguities and uncertain environments. Few NILM algorithms use time-series data for the classification of the appliance.

Some algorithms classify appliances based on time series. In these systems, after the event detection, the ML algorithm performs the classification using as input a raw data window from the time series or characteristics extracted from them [75]-[78].

Finally, unsupervised [48],[74] and semi-supervised learning algorithms [79],[80], graph signal processing [81]-[84], and successive elimination steps with Maximum a Posteriori estimator [85] have been proposed.

2.1.4 Event-based system example: the Hart's algorithm

This algorithm, proposed by George Hart in 1985 [19], is often used as a baseline model for the NILM problem. The overall algorithm can be broken up into eight steps:

1. Measurements
2. Normalization
3. Edge Detection
4. Clustering
5. ON/OFF Matching
6. Separating Simultaneous Changes
7. Transfer to Central Facility
8. Identification.

The author states that each of these steps can be performed in several alternative ways.

In his work he first describes the most suitable way and then shows several options that may be appropriate in certain circumstances.

Measurements

The power and voltage of the buildings are measured with a sampling rate of 1 Hz. A "power vector" is defined containing 4 measurements relating to the real and reactive power measured for the two phases that supply the system. This is mainly because in the United States, the energy is distributed with two 120 V phase cables, 180 ° out of phase with respect to a neutral, and the neutral cable itself.

Normalization

From this data, the real and reactive parts of the power are adjusted every second to correct for the fact that utility allows the line voltage to vary, using the following formula:

$$Adjusted\ Power = Measured\ Power \cdot \left(\frac{120}{Voltage}\right)^2 \tag{2.2}$$

The choice of exponent 2 is due to the assumption that the power is proportional to the square of the voltage. This normalizes the power to what it would have been if the utility voltage were the nominal 120 V. In this way the power variations caused by the line voltage are eliminated.

Edge Detection

The third phase of the method consists in detecting variations in the power, using the following procedure consisting of two phases, illustrated in Figure 2.1 for hypothetical power measurements:

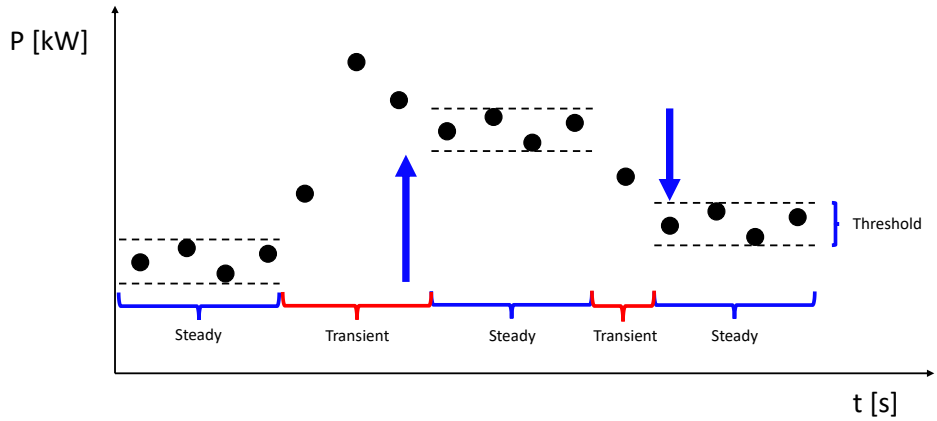

Figure 2.1: Edge detection.

The system divides the sequence of power measurements into time periods where power is stable and in others where it changes. A steady period is defined to be one of a certain minimum length in which the load does not vary more than a specified tolerance. The remaining periods, in between the steady periods, are defined to be the periods of changes. The current embodiment of the devices uses two seconds as the minimum length and 15 W or VAr as the allowable tolerance in the definition of a steady period. Note that a time period is defined to be steady if, and only if, all the measured quantities in the measurement vector remain steady. If any of the components are changing, the period is "changing".

For each time period in which the power is changing, the system calculates the total power change over the period by subtracting the constant power level detected before the start of the change from the constant power level at the end of the change. The effect of noise is reduced by averaging all measurements (of each vector component) during each steady period, to obtain noise-reduced steady values. The change, or transition, for each period of change is therefore a four-component vector, computed by subtracting the average of all measurement vectors in the previous steady period from the average of the measurement vectors in the subsequent steady period.

It is important to note that this description is only appropriate if a long stream of four-component measurements at one-second intervals can be stored in the measurement system, referred to as the Load Monitor. The Load Monitor prototype described by the author uses an edge detection algorithm which is a dynamic version of the static description just described. It produces the same effect by means of a small number of sufficient statistics, without the need to memorize a long stream of measurements.

The dynamic Edge Detection algorithm uses a buffer structure that calculates the average of the measured electrical power. Every time a new sample (at a sampling frequency of 1 Hz) is acquired, the average is updated using the (2.3) that takes into account both the new sample and the previous average.

$$Average = \left(\frac{N}{N+1}\right) \cdot Average + \left(\frac{1}{N+1}\right) \cdot Measurement \qquad (2.3)$$

If a change in power exceeds a pre-set threshold value, it marks the end of a steady state and the beginning of a transient state, which is used to identify the use of a specific appliance. This process continues with the constant updating of the average with new samples.

Clustering

The detected changes are subsequently grouped into "clusters". A change set is simply a set of changes, all roughly equal (in all components). For example, Figure 2.2 shows a hypothetical one-dimensional example in which many changes have been observed and which can be grouped into four clusters, relative to the measured electrical power. Each change is approximately 200 W, 500 W, -200 W or -500 W.

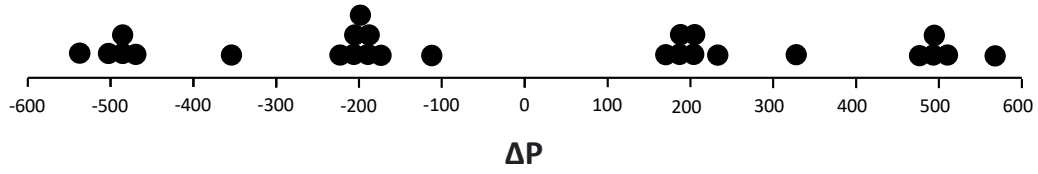

Figure 2.2: Example of clusters.

Cluster analysis allows for some tolerance in the variation measured each time an appliance is powered on or off. For example, the data in Figure 2.2 could refer to a residence with a 200 W and a 500 W appliance. Each time the first appliance is switched on, the total power increases by approximately 200 W (not necessarily exactly by 200 W). This is due to both changes in conditions when the unit is turned on and noise in the metering system. Similarly, the cluster of variations of about 500 W results from the times that this device turns on. Change groups with negative power levels result from switching off the devices.

In general, the real data is more complex than shown in the simple example above. There are likely to be several dozen clusters, because typically in a residence there are dozens of household appliances. However, there are more types of information available than in this one-dimensional example. The independent components of the transition vectors allow grouping into a larger number of dimensions. For example, it is possible to adopt a four-dimensional clustering to separate equipment that draws the same real and reactive power, but are located on different phases. The clustering technique used in the load monitoring prototype has several new features that allow it to function recursively in dynamic implementation.

ON/OFF Matching

The observed changes between the ON and OFF clusters of each appliance are grouped into pairs, based on their time coordinates. Each ON/OFF pair corresponds to a single cycle of appliance usage. For example, if there is a change of about 200 W at 6:00 and a negative change of about 200 W at 9:00 (with no other changes

by any of these clusters in the meantime), they are grouped into a single appliance cycle. From this the algorithm calculates that the 200 W appliance has been on for three hours and consumed 600 Wh of energy. Changes that are not part of an ON / OFF toggle are ignored unless they can be detected by the method described in the next section.

Separating Simultaneous Changes

From time to time two appliances may turn on or off at the same time (or one turns on and one off), or in rapid succession so that the second appliance is turned on before the transient of the first is over. When this occurs, the variation calculated with the previously described method (Edge Detection) will be the sum of the variations that would have been observed if the two appliances had been switched at different times. For example, if the 200 W and 500 W appliances are switched on almost simultaneously, a 700 W increase in the total energy consumption of the house is observed.

This 700 W change is easily interpreted by the facts that:

A. It rarely happens (e.g. the cluster of 700 W changes is very small; perhaps containing only one example).
B. It occurs, in time, between two ON or two OFF transitions of some appliances, which could not both be paired by the matching procedure above (eg. if the 700 W change occurs between two -200 W changes in a row, probably a change of +200 W is missing, similarly if the change of 700 W starting between two changes of -500 W, a +500 W is missing).
C. The observed change is approximately the sum of the two missing changes (e.g. 700 = + 200 + 500).

When all three of these conditions occur, the unusual observed transition is "broken apart" into its two simultaneous components, and the procedure continues as if the two components were available for matching ON's and OFF's as above.

Thus the load monitor "understands" that the 700 W change was really two independent appliance transitions which happened to occur at the same moment.

Transfer to Central Facility

The final step of the method is to output the characteristics of the observed devices. This includes a description of the clusters in the signature space, and parameters that specify their electrical power consumption such as their total power consumption. The energy for any appliance during a given hour is simply the sum of the energy consumed in each of the observed cycles (as calculated in the ON / OFF correspondence section) that occurred during the specified hour.

Identification

Each pair of ON / OFF clusters (a positive and negative cluster of the same size) represents a separate two-state appliance or appliance component (e.g. the heater and motor components of a dishwasher can be viewed as two ON clusters / OFF separate). The algorithm must examine the properties of the clusters and try to identify the class of household appliances of each (e.g. "refrigerator", "dishwasher heater" etc.). To do this, a table of appliance classes and their properties will be provided. The algorithm will check each cluster against the classes in the table to see which element in the table is closest to each observed cluster pair.

The properties used will include real and reactive components of the turn-on transitions. For example, refrigerators as a class are expected to be approximately 4000 W. Weather related correlation factors can also be included, through the introduction of appropriate sensors. Space heating can be identified by the fact that it comes on more frequently when it is cold outside. Air conditioners should be identifiable by their positive correlation with temperature.

The table will also contain timing information, such as the average length of time per ON/OFF cycle of the appliance and the number of cycles per day. Expected time-of-day and time-of-year properties can also be used (e.g. lights are used more often at night, electric lawn mowers are used more often in the day and in the summer).

However, Hart did not wish to rely too heavily on temporal expectations, as this could cause the load monitor to fulfill its own prophesies, and find only predictable results. For example, if the table erroneously claimed that lights are only used in the evening, and someone has turned on the lighting all day, there would be a danger of the load monitor misidentifying the lights and calling them by another name.

2.2 Non-event based methods

Non-event-based NILM systems have been introduced in the last decade, so they can also be considered as an evolution of the other systems. They use as input a window of samples of the aggregate signal (therefore time series data), which are processed continuously, without waiting for the occurrence of events. For this reason, this type of system is particularly suitable for low frequency measured signals. Indeed, it was developed precisely to allow the processing of signals acquired with reduced frequencies, for which the detection of events is more difficult. The processing techniques are therefore different from those previously discussed.

In some cases, the disaggregation problem is formulated as a blind source separation (BSS) problem, that is, the problem of recovering a signal from a set of mixed signals.

This section describes the main approaches used for the implementation of non-event-based NILM systems, as follows:

- Combinatorial Optimization;
- Discriminative Sparse Coding;
- Hidden Markov Model Approaches;
- Deep Learning.

Non-event-based NILM systems have several advantages over their event-based counterpart. Some of the non-event-based NILM methods are in fact independent of the number of appliances that form the aggregate signal. Consequently, the problem of scalability and adaptability to the varying number of appliances in the electrical system is automatically solved. On the other hand, these techniques are relatively less accurate than event-based ones. However, the most recent Deep Learning-based systems have shown to have a generalization capacity on unseen scenarios, such as to make the lower performances in terms of accuracy absolutely acceptable.

2.2.1 Combinatorial optimization

The combinatorial optimization algorithm (CO) [86] is considered as a basic algorithm in the NILM literature. The main assumption in CO is that each appliance can be in a given state (1 of K, where K is a small number), where each state has an associated power consumption. The goal of the algorithm is to assign states to household appliances, in order to minimize the difference between the household aggregate reading and the sum of the energy consumption of the different household

appliances. The time complexity of CO has an exponential relationship to the number of devices and therefore does not fit well.

The total load clearly depends on which appliances are switched on at any given time. Assuming that there are *n* appliances, numbered from 1 to *n*, and let *a(t)* be an *n*-component Boolean vector describing the state of the *n* switches at instant *t*:

$$a_i(t)\begin{cases}1, & \text{if appliance } i \text{ is on at } t\\ 0, & \text{if appliance } i \text{ is off at } t\end{cases} \tag{2.4}$$

for $i = 1 \ldots n$.

the vector *a(t)* modulates the power consumption of the individual appliances.

A multiphase load with *p* phases can be modeled as a *p*-vector in which each component is the load on one phase. The total load *p*-vector is the sum of the individual appliance load *p*-vectors for those appliances switched on at any given point in time. This will be a vector function of time that steps in characteristic increments each time an appliance switches on or off. For $i = 1 \ldots n$, let P_i be the *p*-vector of the power that the *i*-th appliance consumes when it is operating.

For the two-phase circuit of Figure 2.3, each P_i is a two-component complex vector. The real and imaginary parts for the complex power in the *j*-th component of the vector correspond to the real and reactive power consumed on the *j*-th phase. One of the two components is zero for 120 V appliances, as only one phase is involved; the two components are equal for balanced 240 V appliances; and an arbitrary vector represents an unbalanced 240 V appliance. Then we model:

$$P(t) = \sum_{i=1}^{N} a_i(t)\, P_i + e(t) \tag{2.5}$$

where *P(t)* is the vector of the aggregate power consumption measured on the *p* phases and *e(t)* is a small noise or error term.

The model suggests a straightforward criterion for estimating the state of the individual appliances: if the *p*-vectors for each of the *n* appliances are known, the measure of the aggregate power *P(t)* is known, the *n*-vector *a(t)* is the vector that minimizes the vector *e(t)* at time *t*. This is a combinatorial optimization problem:

$$\hat{a}(t) = \arg\min_{a} \left| P(t) - \sum_{i=1}^{N} a_i\, P_i \right| \tag{2.6}$$

This problem is computationally problematic and cannot be expected to be solved exactly, except with exhaustive techniques which are impractical, unless the number of devices n is very small. However, it is possible to devise heuristic algorithms that provide reasonable solutions most of the time.

Although mathematically attractive, there are a number of difficulties in estimating $a(t)$ with this algorithm. The fundamental problem with this approach is that the complete set of P_i are never known. In fact, it is not clear whether a residence should be modeled as having a well-defined number n of appliances, because appliances come and go due to purchases, visitors, seasonal changes, etc. If this algorithm were used in the presence of unknown devices, it would mistakenly attempt to describe their behavior as a combination of other known devices. Besides, a small change in the measured $P(t)$ would often be analyzed as a large change in the switching process, $a(t)$, with a number of appliances turning on or off simultaneously, such that the net change approximates the observed change the best possible.

For example, suppose a residence contains four loads of sizes P1 = 100, P2 = 200, P3 = 300, and P4 = 401 W.

If the measured total load at time t is 500 W, the best estimate is that the second and third appliances are on, i.e., $a(t) = [0,1,1,0]$, as that uniquely gives $e(t) = 0$. If a moment later, at time $t + \Delta t$, the measured load increases slightly to 501 W, the best estimate would then be $a(t + \Delta t) = [1,0,0,1]$, which again has $e(t + \Delta t) = 0$, but implies that every appliance changed state in a short interval Δt. The intuition that every appliance in a residence could not change state simultaneously reflects the knowledge of the physical independence of different appliances.

This suggests the following criterion which is not described in the model (2.5).

Switch Continuity Principle: In a small amount of time, only a small number of appliances is expected to change state in a typical load.

Unfortunately, it is rather difficult to quantify this principle in a meaningful way that would lead to an improvement to (2.6). Perhaps (2.6) could be modified for NILM applications by adding a right-hand term proportional to the number of state changes in $a(t)$.

The Hart's proposed algorithm addresses the problem as follows: given the timing and size of the step changes, it is possible to examine a given list of P_i (and the P_i negatives) to determine which appliance turned on (or turned off) causing any changes. This is the essence of this NILM algorithm, with further refinements to be discussed below. Note that this approach does not suffer from the problem described above regarding the example of increasing power from 500 W to 501 W. A variation of 1 W is too small to be considered a step change.

The method is not even confused by an incomplete set of P_i. By specifying a tolerance condition for the match, the system simply ignores any observed changes that are not close enough to any of the given P_i. In this way, a list of the devices of interest to be monitored can be provided and all other activities are ignored. This model is still somewhat simplistic, however, and can be improved in many ways, to handle simultaneous state changes of more than one appliance, non-power signatures, multi-state appliances, etc.

A problem, that can only be partially solved, is that household appliances with electrically identical consumption cannot be distinguished. For example, it may not be possible to separately totalize the power consumed by two 1200 W resistive appliances on the same phase, for example a toaster and a quartz space heater.

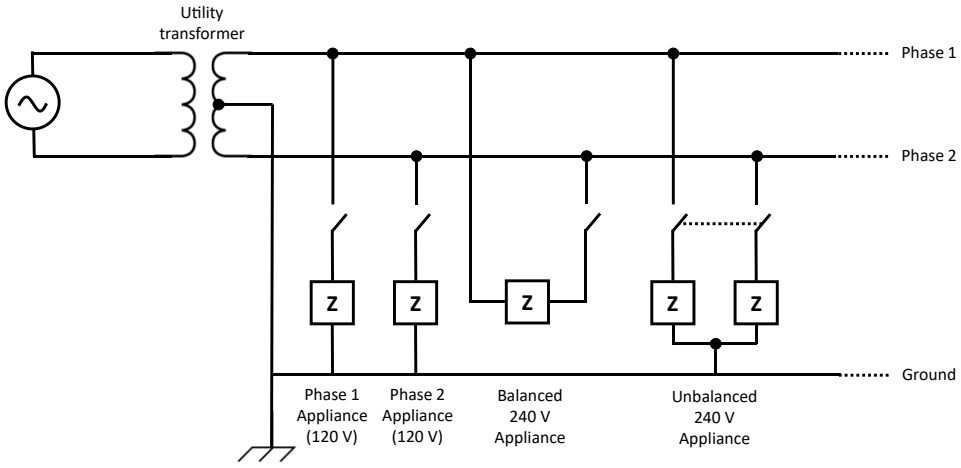

Figure 2.3: Typical U.S. residential electrical system.

To take into account the presence of multi-state appliances, it is necessary to redefine the model.

At a given time, an appliance can only be in a single state, expressed mathematically as:

$$\sum_{k=1}^{K} z_{t,k}^{n} = 1 \tag{2.7}$$

The power consumption by *n*-th appliance in *k*-th state at time *t* is given by:

$$\hat{\theta}_{t,k}^{n} = \sum_{k=1}^{K} z_{t,k}^{n}\, \mu_{k}^{n} \tag{2.8}$$

where $\hat{\theta}_{t,k}^{n}$ is the measured power sequence for *n*-th appliance and μ_k^n is the power draw by *n*-th appliance in *k*-th state.

The overall power consumption of all appliances at a given time *t* is given by:

$$\hat{x}_t = \sum_{n=1}^{N} \sum_{k=1}^{K} z_{t,k}^{n} \, \mu_k^n \tag{2.9}$$

The error in power signal (unaccounted power) after the load assignment is given by:

$$e_t = \left| x_t - \sum_{n=1}^{N} \sum_{k=1}^{K} z_{t,k}^{n} \, \mu_k^n \right| \tag{2.10}$$

Combinatorial optimization seeks to find the optimal combination of appliances in different states which will minimize this error term, using the following state assignment scheme:

$$z_t = \arg\min_{z_t} \left| x_t - \sum_{n=1}^{N} \sum_{k=1}^{K} z_{t,k}^{n} \, \mu_k^n \right| \tag{2.11}$$

where z_t is the set of *n* boolean vectors z_t^n, each of these vectors has a number of elements equal to the number of possible states for the *n*-th appliance and has only one element different from 0 (and equal to 1) corresponding to the state in which the *n*-th appliance is at time *t*.

The corresponding predicted power draw by *n*-th appliance is given by:

$$y^n = \left\{ \mu_{z_1^n}^n, \ldots, \mu_{z_T^n}^n \right\} \tag{2.12}$$

The state space size of this optimization function is K^N, implying that it is exponential in the number of appliances.

Batra et al. [87] described an Improved NILM using load Division and Calibration (INDiC), which provides preprocessing procedures that can simplify NILM computation and improve the overall disaggregation accuracy. These procedures can basically be classified as data cleansing (time series synchronization, downsampling and calibration) and problem division into subproblems (network load assignment). They presented INDiC-CO (INDiC using

Combinatorial Optimization for NILM) but INDiC can be used with any NILM approach.

Load division

Across many countries, electrical distributions are planned such that different loads are connected to different phases.

Since implementing NILM typically involves separate monitoring of different electrical phases, load division can be applied to perform efficient disaggregation. Considering *p* phases in a house, one can operate by first performing the automated assignment of the loads (for a total of *N* loads in the house) to the individual phase (with the result that the N_i loads are assigned to the *i*-th phase).

This division of the load between different phases results in an exponential reduction of the state space for the disaggregation of each phase separately (the size of the state space for the *i*-th phase is given by K^{Ni}). The CO formulation for the *i*-th phase after load division is given by the following optimization function:

$$z_t = \arg\min_{z_t} \left| \theta^{M_i} - \sum_{n=1}^{N} \sum_{k=1}^{K} z_{t,k}^{n} \, \mu_k^n \right| \qquad \forall \quad i \in \{1, \dots, p\} \tag{2.13}$$

where *p* is the number of phases and θ^{M_i} is the measured power sequence for the *i*-th phase.

The corresponding predicted appliance power sequence for *n*-th appliance is given by:

$$y^n = \left\{ \mu_{z_1^n}^n, \dots, \mu_{z_T^n}^n \right\} \tag{2.14}$$

CO with load division is subject to the following constraints:

1) The sum of number of loads assigned to different mains must be equal to the total number of loads. This is given by: $\sum_{i=1}^{p} N_i = N$.
2) At any given time, an appliance can only be in a single state which is given by:
   $\sum_{k=1}^{K} z_{t,k}^{n} = 1$.
3) An appliance can belong to one and only one phase.
4) The sum of power consumption of all appliances assigned to *i*-th phase is always lesser than or equal to the total power of the phase (i.e. e_t term for *i*-th phase will be non negative).

Time series synchronization

Aggregate power and appliance power are typically measured using different hardware. As an example, in REDD [88] TED meters are used to measure the aggregate power and Power House Dynamics are used to measure appliance circuits. It is so possible that some hardware malfunctions occur during the data collection process or that the data is not properly aligned over time. In this step we make sure that the mains power and appliance power time series start and end at the same time. Additional missing data is handled using techniques such as forward filling (padding).

Downsampling

When running CO, transients and fluctuations in the power signal need to be filtered out. Transients occur due to the high starting current of the appliance, while fluctuations are a consequence of the small voltage fluctuations and oscillatory nature of the appliances. Figure 2.4 show both filtering of starting current (left) and voltage fluctuations (right) by downsampling. Filters such as moving mean / median can be used to sample a time series in a time window, whereby the value assigned to the filtered series for a time window is the mean / median of the original series occurring during that time window.

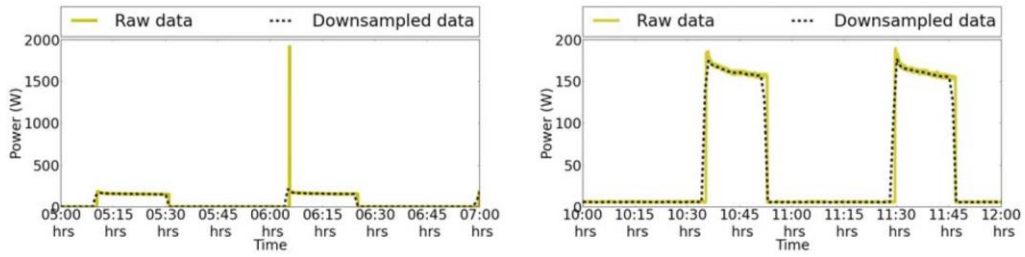

Figure 2.4: Effect of downsampling appliance data.

Assigning Loads to phases

This step aims to identify the mapping between appliances and phases. Since an appliance can belong to a single phase, the mapping is a one-to-one function. Since patterns corresponding to appliances with a higher peak power are generally easier to extract from the phase signal, we first sort the appliance in descending order of peak power.

Starting from the appliance with the highest peak load, one appliance is analyzed at a time and its power is compared at each moment with the power of each of the phases. If the power of the appliance is greater than the power of the appliance

assigned to that phase, the appliance is assigned to the other phase. If the system is unable to assign a fixture to phases using this approach, it finds the times when events occur in the fixture's power series. These events should be a subset of times for the phase this appliance is assigned to. The threshold used to find these events should be properly chosen to ensure that minor voltage fluctuations are not counted as events.

Once an appliance has been assigned to a phase, using one of these two filters, its power supply sequence is subtracted from the corresponding phases, to simplify the assignment of the network to the other appliances. Figure 2.5 (left) shows the assignment of the refrigerator to phase 2, since during this time the refrigerator power is greater than that of phase 1. It can also be seen that the events in phase 2 and the refrigerator power series occur simultaneously.

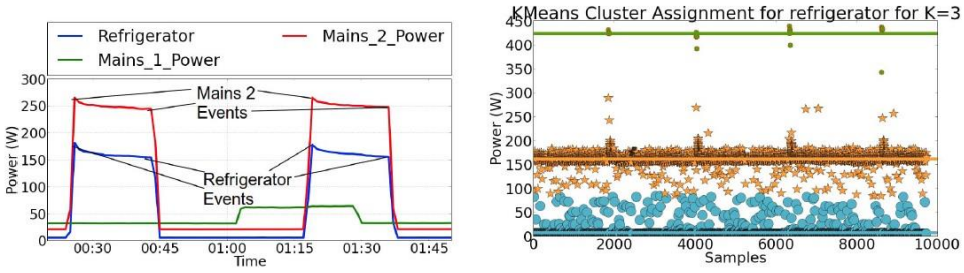

Figure 2.5: Load Assignment and Clustering.

Clustering

The number of states associated with a device (appliance) are identified on the basis of previous knowledge relating to its internal structure [89]. For example, a refrigerator is a compressor appliance and has three states in ascending order of power request (compressor off, compressor on, defrost mode). The corresponding cluster assignment is shown in Figure 2.5 (right).

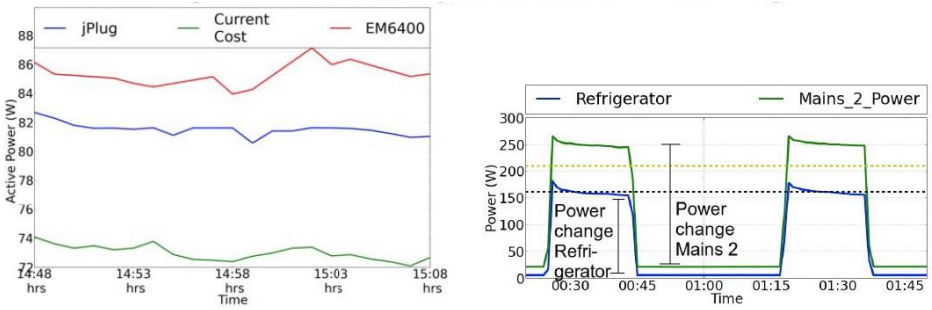

Figure 2.6: Need for and utility of Calibration.

Appliance Power Calibration

Power measured by appliance level meters may need calibration due to several reasons, including:

- Difference in the measurement devices can result in different measurements for the same appliance. To illustrate this difference, the authors measured their refrigerator power with 3 different devices: i) jPlug; ii) Current Cost CT; and iii) EM6400 smart meter. Figure 2.6 (left) illustrates the power measurement by each of these devices. There is a difference in approx. 10 W in measurements reported by jPlug and Current Cost. jPlug gives comparable results to EM6400.
- Voltage fluctuations from the grid resulting in power measurement fluctuations.
- Missing meta data - labeling the appliance level power consumption as real or apparent power.

Motivated by such requirements, INDiC introduces measurement calibration. In comparison to appliance data, mains data is usually measured with better precision devices. Thus, they keep mains data as a reference and calibrate appliance data against it. In the clustering step, value of appliance power at each time is associated with corresponding cluster state ($k \in \{1, \cdots, K\}$). Since in off state (k =1) appliance power consumption is almost zero, it does not require any calibration.

The times of the events are identified by detecting the instants when the appliance switches from a lower state (k) to a higher state (k + 1). In these instants the ratio between the entity of the power variation that occurs in the assigned mains and in the appliance is calculated. This ratio serves as a corrective multiplication factor for a particular state of the appliance. The cluster centroids obtained in the previous step are multiplied by this factor to obtain the calibrated cluster centroids.

This process is shown in Figure 2.6 (right), where it is observed that the refrigerator power in state 2 of 162 W was changed to 214 W after calibration (with steps 2), with a calibration factor of 1.34.

By following these steps, combinatorial optimization is performed separately for both phases as described above.

2.2.2 Discriminative sparse coding

To examine sparse coding methods and their application to disaggregation activities, the terminology of the energy disaggregation domain will be used, although the algorithms can equally apply to other domains as well. Formally, it is assumed that *k* different classes are defined, which in the present case correspond to categories of devices such as televisions, refrigerators, heaters, etc.

For every $i = 1,\ldots, k$, a matrix $\boldsymbol{X}_i \in \mathbb{R}^{T\times m}$ is defined where, where each column of $\boldsymbol{X}_i$ contains one week of energy consumption (measured every hour) for a particular house and for that particular type of device. Thus, for example, the *j*-th column of $\boldsymbol{X}_i$, denoted as $x_1^{(j)}$, can contain the weekly energy consumption for a refrigerator (for a single week in a single house) and $x_2^{(j)}$ could contain the weekly energy consumption of a heater (for this same week in the same house).

Therefore assuming that the matrix $\boldsymbol{X}_j$ is that relating to the fridge, we will have:

$$\boldsymbol{X}_j = \boldsymbol{X}_{fridge} = \begin{bmatrix} x^{(fridge)}_{house\ 1\ (00:00)} & \cdots & x^{(fridge)}_{house\ m\ (00:00)} \\ \vdots & \ddots & \vdots \\ x^{(fridge)}_{house\ 1\ (23:00)} & \cdots & x^{(fridge)}_{house\ m\ (23:00)} \end{bmatrix} \tag{2.15}$$

The aggregate power consumption on all types of devices is indicated as $\bar{\boldsymbol{X}} = \sum_{i=1}^{k} \boldsymbol{X}_i$ so that the *j*-th column of $\bar{\boldsymbol{X}}$, $\bar{x}^{(j)}$, contains a week of aggregated energy consumption for all devices in a given house. At training time, it is assumed to be able to access the energy readings of the single device $\boldsymbol{X}_1, \ldots, \boldsymbol{X}_k$ (obtained for example from plug-level monitors in a small number of instrumented homes). At test time, however, it is assumed to access only to the aggregate signal of a new set of data points $\bar{\boldsymbol{X}}'$ (as would be reported by smart meter), and the goal is to separate this signal into its components, $\bar{\boldsymbol{X}}'_1, \ldots, \bar{\boldsymbol{X}}'_k$.

The sparse coding approach to source separation [90], [91], which forms for the basis for this disaggregation approach, is to train separate models for each individual class $\boldsymbol{X}_i$, then use these models to separate an aggregate signal. Formally, sparse coding models the *i*-th data matrix using the approximation $\boldsymbol{X}_i \approx \boldsymbol{B}_i\boldsymbol{A}_i$ where the columns of $\boldsymbol{B}_i \in \mathbb{R}^{T\times n}$ contain a set of *n* basis functions, also called the dictionary, and the columns of $\boldsymbol{A}_i \in \mathbb{R}^{n\times m}$ contain the activations of these basis functions.

Sparse coding additionally imposes the constraint that the activations $\boldsymbol{A}_i$ are sparse, i.e., that they contain mostly zero entries, which allows us to learn overcomplete representations of the data (more basis functions than the dimensionality of the data).

A common approach to achieving this sparsity is to add an ℓ_1 regularization penalty to the activations.

Since energy usage is an inherently non-negative quantity, we impose the further constraint that the activations and bases be non-negative, an extension known as non-negative sparse coding [92],[93]. Specifically, we will consider the non-negative sparse coding objective.

$$\min_{\boldsymbol{A}_i \geq 0, \boldsymbol{B}_i \geq 0} \left(\frac{1}{2} \|\boldsymbol{X}_i - \boldsymbol{B}_i \boldsymbol{A}_i\|_F^2 + \lambda \sum_{p,q} (\boldsymbol{A}_i)_{pq} \right) \quad subject\ to \quad \left\|\boldsymbol{b}_i^{(j)}\right\|_2 \leq 1, \quad j = 1, \ldots, n \tag{2.16}$$

where $\|Y\|_F = \sqrt{\sum_{p=1}^{m} \sum_{q=1}^{n} \left|y_{p,q}\right|^2}$ is the Frobenius norm.

We can interpret the first term of the sparse coding objective as a reconstruction term which tries to force the algorithm to provide a good representation of $\boldsymbol{X}_i$ and the second term as a sparsity penalty which forces our representation of $\boldsymbol{X}_i$ to be sparse. The constant λ is a scaling constant to determine the relative importance of these two contributions. Put simply, we're just trying to solve a problem (disaggregation in this case) using as little resources as possible.

The constraint $\left\|\boldsymbol{b}_i^{(j)}\right\|_2 \leq 1$ is necessary as otherwise we could minimize $\frac{1}{2}\|\boldsymbol{X}_i - \boldsymbol{B}_i \boldsymbol{A}_i\|_F^2$ as much as we want, making $\boldsymbol{B}_i$ large enough to compensate. We want to avoid this behavior as we are looking for vectors containing as few nonzero and large values as possible.

After using the above procedure to find representations $\boldsymbol{A}_i$ and $\boldsymbol{B}_i$ for each of the classes $i = 1, \ldots, k$, we can disaggregate a new aggregate signal $\overline{\boldsymbol{X}} \in \mathbb{R}^{T \times m'}$ (without providing the algorithm its individual components). We concatenate the bases to form single joint set of basic functions and solve the optimization problem.

$$\widehat{\boldsymbol{A}}_{i:k} = \underset{\boldsymbol{A}_{i:k} \geq 0}{\arg\min} \left(\frac{1}{2} \left\| \overline{\boldsymbol{X}} - [\boldsymbol{B}_1 \quad \cdots \quad \boldsymbol{B}_k] \begin{bmatrix} \boldsymbol{A}_1 \\ \vdots \\ \boldsymbol{A}_k \end{bmatrix} \right\|_F^2 + \lambda \sum_{p,q} (\boldsymbol{A}_i)_{pq} \right) \tag{2.17}$$

We then predict the *i*-th component of the signal to be:

$$\widehat{\boldsymbol{X}}_i = \boldsymbol{B}_i \widehat{\boldsymbol{A}}_i \tag{2.18}$$

Kolter et al. [94], who first proposed this approach, focused on the disaggregation of electricity using hourly data; they specifically examine the generalization capability of their algorithm using a dataset with 590 houses and power consumption over a period of over two years.

The dataset was provided by Plugwise, a European manufacturer of plug-in monitoring devices. Each device is labeled with one of 52 device classes, which have been further reduced to ten major categories of electrical devices: lighting, TV, computer, other electronic devices, kitchen appliances, washing machine and dryer, refrigerator and freezer, dishwasher, heating / cooling, and a varied category.

2.2.3 Hidden Markov models

In probability theory, a Markov model is a stochastic model used to model randomly changing systems [95]. It is assumed that future states depend only on the current state, not on the events that occurred before it (that is, it assumes the Markov property). Generally, this assumption enables reasoning and computation with the model that would otherwise be intractable (a problem that can be solved in theory but for which in practice any solution takes too many resources to be useful). For this reason, in the fields of predictive modelling and probabilistic forecasting, it is desirable for a given model to exhibit the Markov property.

A hidden Markov model (HMM) is a Markov model in which the sequence is made up of discrete variables. In addition, each discrete variable emits a single continuous variable, which is dependent upon the value of the discrete variable. Furthermore, in a HMM the chain of discrete variables is not observed, while the continuous variables are observed. Figure 2.7 shows the graphical structure of a HMM, where the discrete, hidden variables are represented by the sequence $\boldsymbol{z} = (z_1, \dots, z_T)$, and the continuous, observed variables are represented by the sequence $\boldsymbol{x} = (x_1, \dots, x_T)$, where T is the length of the sequence (the number of time slices in the model). The value of each discrete variable z_t corresponds to one of K states, while each continuous variable can take on the value of any real number.

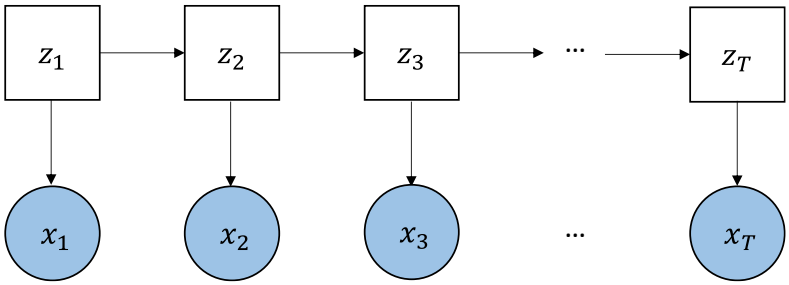

Figure 2.7: Hidden Markov Model.

The behaviour of a HMM can be completely defined by three parameters. First, the probability of each state of the hidden variable at $t = 1$ can be represented by the vector π such that:

$$\pi_k = p(z_1 = k) \tag{2.19}$$

Second, the transition probabilities from state i at t-1 to state j at t can be represented by the matrix $\boldsymbol{A}$ such that:

$$A_{i,j} = p(z_t = j | z_{t-1} = i) \tag{2.20}$$

Third, the emission probabilities for x are described by a function governed by parameters which is commonly assumed to be Gaussian distributed. The probability density of the standard Gaussian distribution (standard normal distribution, with zero mean and unit variance) is often denoted with the Greek letter ϕ. The normal distribution is often referred to as $N(\mu, \sigma^2)$. Thus when a random variable X is normally distributed with mean μ and variance σ^2, one may write:

$$X \sim N(\mu, \sigma^2) \tag{2.21}$$

Therefore, the probability that the hidden state z_t emits the variable x_t can be written as:

$$x_t | z_t, \phi \sim N(\mu_{z_t}, \sigma_{z_t}^2) \tag{2.22}$$

Equations (2.19), (2.20) and (2.22) can be used to calculate the joint likelihood of a HMM:

$$p(\boldsymbol{x}, \boldsymbol{z} | \theta) = p(z_1, \pi) \prod_{t=2}^{T} p(z_t | z_{t-1}, \boldsymbol{A}) \prod_{t=1}^{T} p(x_t | z_t, \phi) \tag{2.23}$$

where the set of all model parameters is represented by $\theta = \{\pi, \boldsymbol{A}, \phi\}$

There exist two common goals when applying a HMM to a real world problem. First, one aim is to infer the model parameters θ given a sequence of continuous variables $\boldsymbol{x}$. Second, another aim is to determine how the model parameters θ and a sequence of continuous variables $\boldsymbol{x}$ can be used to infer the optimal sequence of discrete states $\boldsymbol{z}$. These problems will be referred to as learning (or training) and inference respectively.

Learning in the context of HMMs refers to finding values for the model parameters which best explain the training data. The state of the art in terms of maximum likelihood estimation is the Expectation Maximisation (EM) algorithm (also known as the Baum-Welch algorithm). However, approximate methods are more commonly used due to their support for fully Bayesian inference.

Inference in the context of HMMs refers to finding values for hidden variables which maximise the model's joint likelihood. The Viterbi algorithm is the state of the art in this context, and provides an efficient solution to the problem with complexity that scales linearly in the length of the chain. The Viterbi algorithm considers each time slice in sequence, and evaluates the probability of each transition from the previous time slice to the current time slice. However, only the transition with the maximum probability leading to each state in the current time slice is retained. By propagating the maximum probability of each state forwards through the subsequent time slices, the algorithm guarantees that the most probable sequence of states will be retained. When the maximum probabilities of each of the state has been propagated to the final time slice, the probability of each sequence is known. Since a unique transition leading to each state was retained, a single path exists from the most probable state in the final time slice backwards through each previous time slice representing the most probable joint sequence of states. The complexity of exact inference in a HMM is $O(K^2T)$, where K is the number of states and T is the number of time slices.

The formal steps in the Viterbi algorithm to find the best single sequence of states are as follows:

1. *Initialization*

By defining $\delta_t(z_t)$ the probability of the optimal sequence of states that produced the series of observations $x_1, \dots, x_t$ and ending with the state z_t, we can write that:

$$\delta_1(z_1) = p(z_1, \pi) \cdot p(x_1|z_1, \phi) \tag{2.24}$$

That is, the probability that x_1 is observed at the initial time t = 1 is given by the product between the initial probability $p(z_1, \pi)$ that z_1 occurs and the probability $p(x_1|z_1, \phi)$ that z_1 emits x_1.

2. *Recursion*

$$\text{for } \; 2 \leq t \leq T \; \text{ and } \; 1 \leq z_t \leq N$$
$$\delta_t(z_t) = \max_{1 \leq z_{t-1} \leq N} [\delta_{t-1}(z_{t-1}) \cdot p(z_t|z_{t-1}, \boldsymbol{A}) \cdot p(x_t|z_t, \phi)] \tag{2.25}$$

The observation x_t at time t is known, the hidden state z_{t-1} will be the one that determines the maximum product of the probability $\delta_{t-1}(z_{t-1})$, i.e. the probability of the optimal sequence of states that produce the series of observations $x_1, \dots, x_{t-1}$ and ends with the hidden state z_{t-1}, the transition probability $p(z_t|z_{t-1}, \boldsymbol{A})$ from the state z_{t-1} to the state z_t and the emission probability $p(x_t|z_t, \phi)$.

3. Backward flow

Once the end of the sequence has been reached, the probability $\delta_T(z_T)$, i.e. the probability of the optimal sequence of states that produced our entire series of observations $x_1, \dots, x_T$ and ending with the state z_T, is known.

$$z_T = \underset{1 \le z_T \le N}{\arg\max}[\delta_{T-1}(z_{T-1}) \cdot p(z_T|z_{T-1}, \boldsymbol{A}) \cdot p(x_T|z_T, \phi)] \tag{2.26}$$

Known z_T, the sequence of hidden states can be deduced by going backwards as:

$$z_t = \underset{1 \le z_t \le N}{\arg\max}[\delta_t(z_t) \cdot p(z_{t+1}|z_t, \boldsymbol{A}) \cdot p(x_{t+1}|z_{t+1}, \phi)] \qquad \forall\, t = T-1, T-2, \dots, 1 \tag{2.27}$$

The factorial hidden Markov model (FHMM) is an extension of the HMM in which there are multiple independent Markov chains of hidden variables, $\boldsymbol{z}^{(1)}, \dots, \boldsymbol{z}^{(N)}$, where N is the number of chains. In this model, each observation is dependent upon multiple hidden variables. The graphical model of a FHMM is given by Figure 2.8.

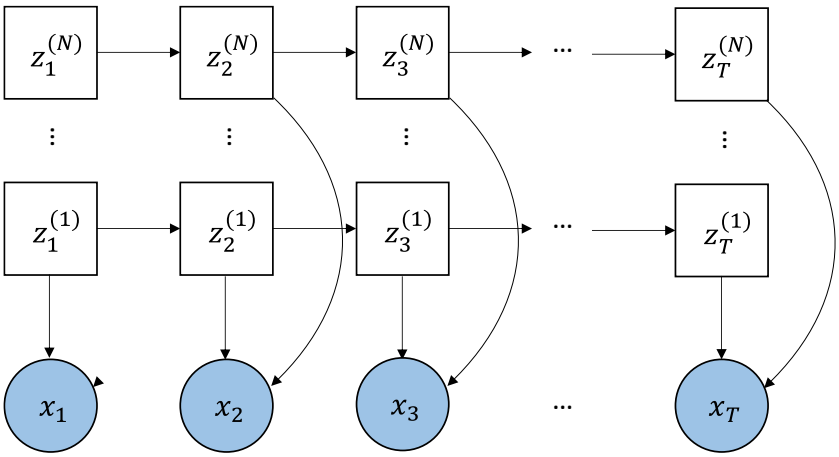

Figure 2.8: Factorial Hidden Markov Model.

Similar to (2.23), the joint likelihood of a FHMM can be calculated by:

$$p(\boldsymbol{x}^{(1:N)}, \boldsymbol{z}|\theta) = \prod_{n=1}^{N} p({z_1}^{(n)}, \pi) \prod_{t=2}^{T} \prod_{n=1}^{N} p({z_t}^{(n)}|{z_{t-1}}^{(n)}, \boldsymbol{A}) \prod_{t=1}^{T} p(x_t|{z_t}^{(1:N)}, \phi) \qquad (2.28)$$

where $1:N$ represents a sequence of appliances 1,…, N.

However, the computational complexity of both learning and inference is greater for FHMMs compared to HMMs. This is due to the conditional dependence of the Markov chains given the observed variables. There are two possible solutions to perform learning and inference in a FHMM. The first is to transform the FHMM into a large HMM and perform learning and inference as discussed above. The alternative is to keep the factorial structure of the graphical model but use approximate techniques for inference. The FHMM can be transformed into an equivalent HMM, which will allow standard HMM inference methods to be applied to the model. This can be achieved by using a single Markov chain with K^N states, one for each combination of states in the FHMM, resulting in a computational complexity of $O(K^{2N}T)$ for exact inference. Since the computational cost is clearly exponential in the number of chains, N, the model will therefore become computationally intractable for large N. Alternatively, approximate methods provide more tractable inference methods.

FHMM is very well suited to the NILM problem, and has found wide use [14],[88],[96]-[103]. Generally, in these works each device is represented by an HMM with K_i states. The state of the i-th appliance is represented by a vector $S_{it} = (S_{it1}, S_{it2}, \cdots, S_{itKi})^T$ where S_{itk}=1 when appliance i at instant t is in state k. All other elements of the vector will be 0. Parameters such as the power drawn by each fixture in k-states, initial probabilities and model transition probabilities will be learned during a training phase. Given the model parameters, the goal is to infer the time sequence of the hidden states S_i for each appliance.

2.2.4 Deep Learning

Machine learning is the field of study that gives computers the ability to learn without being explicitly programmed [104]. Unlike traditional programming, which provides a list of more or less complex rules defined by the programmer in order to obtain certain outputs, machine learning automatically learns patterns and correlations in order to be able to solve extremely complex problems. In problems for which existing solutions require a lot of manual adjustments or long lists of rules, a machine learning algorithm can often simplify the code and achieve better performance. Sometimes they allow to find solutions to problems that otherwise would not be solved through traditional approaches. These algorithms are used to

process large amounts of data in order to discover patterns that are not immediately apparent. They are also used in situations where the algorithm needs to dynamically adapt to new patterns in the data or when the data itself is generated as a function of time, such as stock price prediction, in this case we speak of online learning.

Deep learning (DL) is a type of ML that trains a computer to perform human-like activities, such as speech recognition, image identification, or prediction-making. Instead of organizing data to perform predefined equations [105], DL sets basic parameters on the data and trains the computer to learn on its own, by recognizing patterns using some levels of processing. DL applications use a layered structure of algorithms called the artificial neural network (ANN) since they vaguely imitate the interconnected structure of the human brain to provide multilevel functionality [106], [107].

In the human brain, when a neuron is "activated" it sends an electrical impulse to other neurons along axons. ANNs consist of a network of very simple elements according to a distributed, massively parallel structure capable of learning and thus generalizing. Generalizing means producing outputs at inputs not encountered during training. Due to their structure, DL algorithms have greater generalization capability than traditional ML algorithms.

The fundamental element of an ANN is therefore the artificial neuron, which typically has many inputs and only one output, as shown in Figure 2.9. Each input has an associated weight, which determines the strength of the connection to the next neuron. The activation of the neuron is a function of the weighted sum of the inputs. The ANNs are trained by presenting a set of examples (training set) as input to the network. The response provided by the network for each sample is compared to the desired response. The difference (error) between the two is evaluated and, on the basis of this difference, the weights are adjusted. This process is called back-propagation and is repeated on the entire training set until the network outputs error below a pre-set threshold.

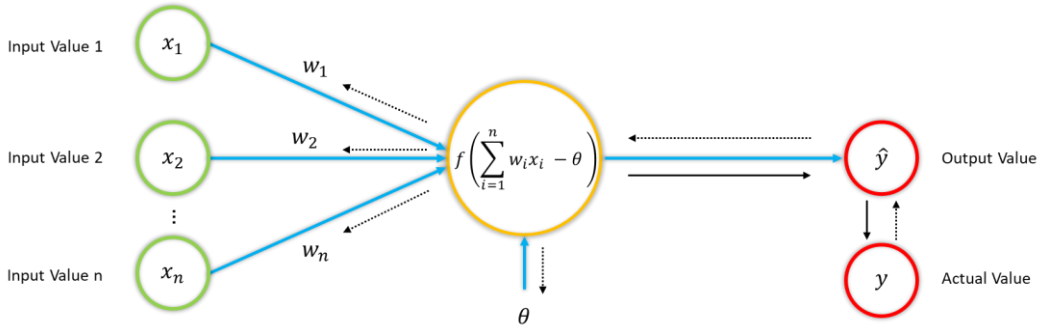

Figure 2.9: Training process for a single artificial neuron.

The output of the neuron, i.e. the signal with which the neuron transmits its activity to the outside, is calculated by applying the activation function to the weighted sum of the inputs. Often, in the literature, the weighted sum of inputs is referred to as "net". A threshold was also included in the neuron model represented in Figure 2.9, which has the effect of lowering the input value to the activation function.

In some cases, instead of considering the threshold, its opposite is considered, called bias, which can therefore be seen as the weight associated with a further input channel with a constant value equal to 1. The bias is indicated as *b* in (2.29).

$$\hat{y} = f\left(\sum_{i=1}^{n} w_i x_i + b\right) = f\left(\sum_{i=0}^{n} w_i x_i\right) \tag{2.29}$$

where $b = w_0$ and $x_0 = 1$.

ANNs implement a feature extraction process. Feature extraction means a set of techniques that make it possible to obtain new features by creating combinations of those already known. These techniques make it possible to reduce the dimensionality of the datasets and discover new more significant characteristics for the purpose of the objective to be achieved, be it a regression or a classification. The extraction is performed by finding the coefficients $w_1, \dots, w_n$ which allow to obtain the new feutures by combining the available feutures together. By incorporating all the neurons into a single model, an ANN is obtained in which each column of neurons represents a layer of the network, each connection has a certain weight and the output of a previous layer will be the input of the next layer.

In general, there is no information about the features of the intermediate layers and it is not even possible to interpret them. The only usable information are those present within the dataset, i.e. the starting features and the target, for this reason the intermediate layers are known as hidden layers.

Since it is not known a priori which features to combine together to extract a new one, each output node of one layer is connected to each input node of the next layer. For this reason, these are called dense neural networks. The training algorithm will then estimate the intensity of each connection and therefore its relative weight.

An ANN can also have more than one hidden layer, in this way the neural network will continue to extract other features from those already extracted reaching an even higher level of abstraction. An ANN with two or more hidden layers is called a deep neural network. Figure 2.10 shows an example of a deep neural network.

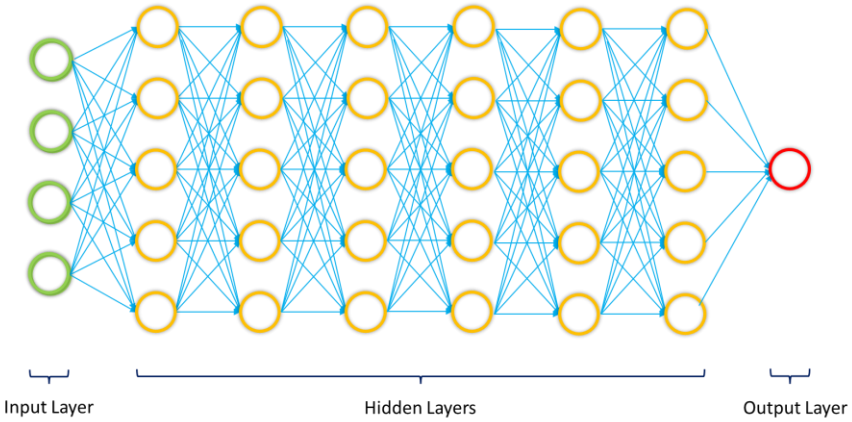

Figure 2.10: Architecture of a deep neural network.

The number of nodes of a hidden layer and the number of hidden layers are hyperparameters to be defined, since it is not possible to know in advance the number of features to extract. The number of nodes in a hidden layer is independent of that of the other paths. The number of hyperparameters to optimize for a deep neural network therefore increases as the number of hidden layers increases.

Back propagation, mentioned above, is the technique by which the ANN parameters are optimized to minimize the cost function. A cost function provides a measure of the distance between the implemented model and the ideal model, i.e. the one that always returns the correct result.

In general, therefore, through the cost function it is possible to evaluate the quality of an ML model. There are different types of cost functions, for regression problems the most common is the sum squared residuals (RSS) [108] while for classification problems it is the Log Loss [109].

Back propagation is based on Gradient Descent, which is the process shown in Fig. 2.9. The learning algorithm, or Back-propagation, performs a number of complete passes of the training dataset. This number is referred to as epochs.

Therefore, each time the entire dataset has passed through ANN, an "epoch" has passed. An epoch can consist of one or several batches. Generally, a data set is divided into a number of batches, and each of these batches has a batch size corresponding to a number of samples.

The batch number defines how many parts the entire dataset will be divided into during the training phase to train ANN on the entire dataset. At the same time, the batch size is the number of samples (training samples) that the algorithm must process on before updating the weights. Thus, the batch size is an indicator of how quickly the ANN weights are updated.

1) *Full Batch Gradient Descend*, in this case all the examples in the training set are used in one batch. At each epoch, the value of the cost function will tend to decrease to a predetermined optimal value.
   This approach has the following limitations:
   - is relatively inefficient for large datasets, as loading the entire dataset into memory can be very computationally expensive;
   - has low dynamics, since to improve the model with new data it is necessary to retrain it on the entire dataset;
   - it is susceptible to the local minima problem.

2) *Stochastic Gradient Descend*, which counteracts the limitations of Full Batch Gradient Descend by running the algorithm on a single example of the training set at a time.
   Unlike Full Batch Gradient Descend, it is most likely that the cost function does not decrease steadily but tends to oscillate. An epoch is completed when the Stochastic Gradient Descend is passed for each example in the training set. At the end of each epoch, it is a good idea to shuffle the examples within the training set to prevent cycles from forming.
   Stochastic Gradient Descend presents several advantages:
   - weighs little in the memory, as it is sufficient to load only one example at a time;
   - has high dynamics, as to update the model it is possible to perform a SGD pass only on the new data;
   - due to fluctuations in the cost function it is less sensitive to the problem of local minima.

   In contrast, it has excessive fluctuations of the cost function, the global minimum point is likely to be missed.

3) *Mini Batch Gradient Descend*, consists of running one step of the Gradient Descend on a given number of examples of the training set at a time (batch size).
   Again, the epoch ends when the algorithm is passed for each example in the training set. As with Stochastic Gradient Descend, the cost function tends to fluctuate but in a smaller way. Recommended batch size values range from 32 to 512, moving by powers of 2.

In addition to densely connected ANNs, there are two other main classes of ANNs:

- *Convolutional neural networks (CNN)*, widely used for computer vision applications. They exploit a process mathematically known as "convolution." In mathematics, convolution is an integral that expresses the superposition of one function over another. These particular ANNs analyze the input through a series of filters known as kernels. Kernels are matrices (in the 2-D case) that shift over the input data, used to extract features from images.
- *Recurrent neural networks (RNN)*, capable of creating an internal network memory that allows the output of one execution to be passed to the next by creating a loop that connects the same hidden layer between different executions, as shown in Fig. 2.11. An RNN processes sequences by iterating the elements of the sequence and maintaining a state that contains information about what it has seen previously.

  When RNNs are trained on very long sequences, they run into two problems typical of these architectures: vanishing gradients and exploding gradients. The former occur when the gradients used to calculate the weights start to vanish, i.e. they become small numbers close to zero. As a result, the network does not learn. The opposite problem is that of exploding gradients. This problem is solved by cutting the gradients. The vanishing gradient problem is solved by particular types of RNNs, namely Long-Short Term Memory (LSTM) and Gated Recurrent Units (GRU).

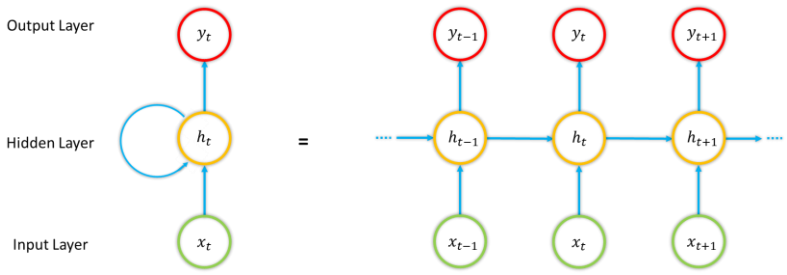

Figure 2.11: Structure of a RNN.

Although densely connected ANNs have been widely proposed as load signature classifiers in event-based systems, their main contribution has been in the development of non-event-based systems.

The first to propose such an approach were Kelly and Knottenbelt [110], using three different types of neural networks. The most significant results are those shown by the architecture the authors refer to as Denoising AutoEncorder (dAE). AutoEncoders (AEs) are neural networks trained to reconstruct their input by passing it through a compact vector representation. The input is then first encoded and then decoded to be reconstructed.

The dAEs are AEs that attempt to obtain a clean target from a noisy input, as may be the case with removing reverberation from an audio recording, or removing grain from an old photograph. In the case of NILM, the energy disaggregation can be seen as a denoising problem, in which one tries to recover the "clean" signal of the power profile absorbed by the single appliance from the background "noise" consisting of the absorptions of the other appliances.

Kelly and Knottenbelt's also introduced the concept of sequence-to-sequence architectures for energy disaggregation, that is, networks that map an output sequence from an input sequence of identical length. By adopting sliding window processing, such an approach leads to multiple predictions for a single timestamp. Therefore, the mean of the predictions is considered as the result. Subsequent to their work, other researchers have proposed similar architectures in order to improve the obtained performance [111]-[113], including trying to use reduced lengths for the output sequence, thus defining a sequence-to-subsequence approach [114]-[128]. A significant breakthrough was achieved when Zhang et al. proposed a sequence-to-point architecture [113], assuming that the midpoint element of the sequence can be represented as a nonlinear regression of the input window.

These architectures generally consist of feedforward elements, that are special convolutional elements. In fact, the latter are used about twice as often as recurrent elements. In recent years, elements such as dilated convolutions [129]-[133], the attention mechanism [134]-[136], generative adversarial networks (GANs) [134],[137]-[140] have also been adapted in architectures for NILM.

Chapter 3

Applications in industrial and residential context

Research on NILM systems, in addition to focusing on theoretical models, also focuses on the large-scale implementation of these systems. In this context, other application fields have been identified, in addition to the classic one previously described. Access to information relating to the energy consumption of individual household appliances certainly helps consumers to have a greater awareness of bill costs. This allows them to improve their behavior in terms of optimizing consumption. Studies have shown that the potential bill savings for consumers who are provided with detailed information on the consumption of individual appliances can exceed 12% [141].

However, the potential applications of NILM systems in the residential and industrial fields are not limited to information aimed at saving energy. This chapter describes the most concrete applications in which NILM systems have been successfully employed in recent years:

1) *Energy tariffs recommendation systems*: By analyzing energy consumption patterns and the usage of individual appliances, energy providers can offer customized tariff plans to customers, resulting in cost savings for both the consumer and the provider.
2) *Microgrid & Smart Home Energy Management Systems*: NILM can also be integrated into energy management systems for Smart Homes and Microgrids. These systems can provide real-time monitoring and control of energy consumption, allowing for efficient distribution of energy resources and reduction of waste.

3) *Demand Response in Smart Grids*: Energy disaggregation system can play a crucial role in Demand Response systems, which are used to manage and balance energy demand on a larger scale. By accurately predicting energy consumption patterns, NILM can help to prevent overloading of the grid and ensure a stable energy supply.
4) *Anomaly Detection and Maintenance*: By monitoring energy consumption patterns of individual appliances, NILM systems can detect abnormal usage and alert the user to potential issues, allowing for proactive maintenance and reducing the risk of equipment failure.
5) *Disaggregation of regional demand*: This technology can be applied at the electrical substation level for energy consumption disaggregation. This can provide valuable insights for distribution network operators, allowing them to better understand and manage the status of the network.
6) *Ambient Assisted Living*: NILM systems can provide real-time monitoring of daily activities and enable personalized care and support for elderly or disabled individuals.

It is easily conceivable that these techniques could also find other types of more innovative applications in the future.

3.1 Energy tariffs recommendation systems

The detail of domestic energy consumption is still largely invisible to millions of users and this is one of the main causes of much waste. Consumption feedback is necessary for energy saving, even if it is not always sufficient because it is difficult to understand for most consumers.

A direct feedback could therefore be extremely valuable, especially for the savings deriving from the daily use of household appliances. In the longer term and on a larger scale, informative billing and annual energy reporting can promote investment and influence behaviour. Savings of the order of 5-15% and 0-10% have been recorded for direct and indirect feedback, respectively [142]. The direct (immediate) feedback refers to instantaneous measurements of power or energy and their communication. Indirect feedback, on the other hand, includes all the information deriving from the periodic processing of energy consumption, such as for example the comparison with consumption periods other than the one recorded, the comparison with other households or with a target indicated to the user via monitor.

The potential contribution of a NILM system in the home is therefore not negligible. An instant and easily accessible display can provide the consumer with adequate information on the various appliances, showing the increase in consumption due to kettles, ovens, refrigerators, washing machines, dishwashers, etc. . Information on the contribution of individual consumption to the total energy consumption of a home could also be provided on the bill, as a general guide.

A display that shows information on energy use is very useful at home, for viewing instantaneous consumption, the cost over the period and the history, with the option of also viewing information on tariffs.

Feedback on consumption is a valuable learning tool and must be seen within the whole context. Its impact varies according to the circumstances, but can sometimes also be improved by accompanying feedback with advice and information, as shown in Fig. 3.1.

Fisher et al. [143] developed a customized system of energy recommendations. The system connects to an energy rate comparison website and estimates annual costs based on personal usage profiles.

Through a NILM algorithm, the deferrable loads (washing machine, dryer, dishwasher, etc.) are detected by measuring the aggregate power. Based on user profiles, the system provides comparisons of the user's current energy tariffs to the tariffs available on the market, and provides advice on how much the user can save by shifting detected deferrable loads to off-peak times. The system estimates the

annual costs on the basis of the consumption forecast obtained from the measurements collected and on the basis of the various tariffs available. Knowing the contribution of deferred loads and their periods of use, the system also makes forecasts on the basis of the shift of these loads at time bands subject to cheaper prices. During the study, the system found cheaper tariffs for 9 out of 10 participants, with an estimated annual saving of £35 to £391 by switching to a cheaper tariff. In addition, the system calculated additional savings of between £26 and £110 by switching 20% of deferrable loads to the cheapest nightly rate.

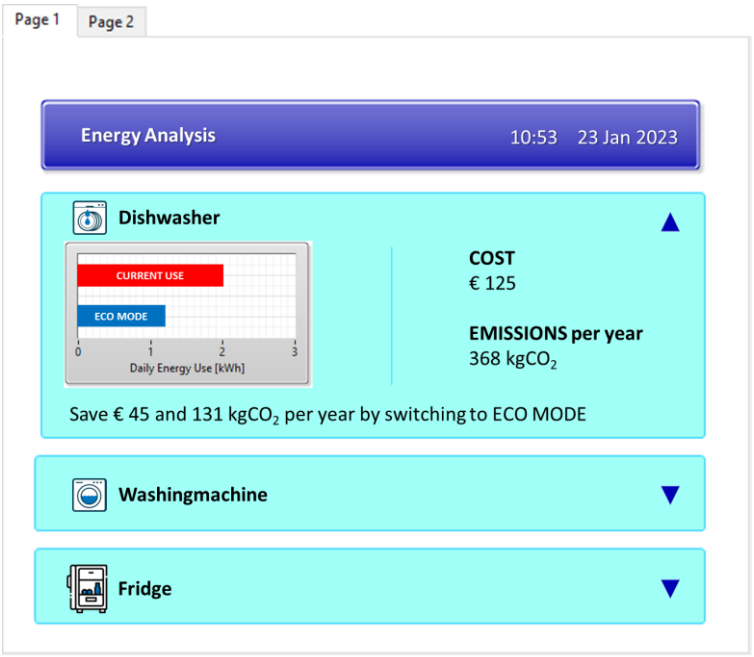

Figure 3.1: User interface of an energy recommendation system.

3.2 Energy management systems

An EMS allows the user to measure and control the use and production of energy within a plant. Due to the benefits offered to both the consumers and distributors of electricity, EMSs are becoming increasingly important for reducing consumption. The most demanding sector, from the point of view of electricity consumption, is represented by buildings, which consume 40% of the total primary energy [144]. An EMS first and foremost analyzes the electrical energy consumption of appliances in order to schedule their use. In addition, it manages the flow of electrical power from renewable sources and storage when they are locally available.

3.2.1 EMS for smart homes and microgrids

NILM techniques allow for the real-time detection of devices that are active. They can be used within EMSs, providing information on the electrical consumption of individual appliances. To apply these techniques advantageously, it is firstly important to classify the load devices on the basis of the programmability or, otherwise, of their use. The former are devices whose operation cannot be delayed (e.g., lighting, kitchen, or refrigerator). The latter devices can operate in variable time intervals based on the price of energy (e.g., washing machines and dryers). Heating, Ventilation, and Air Conditioning (HVAC) systems are particularly important to this aim.

The EMS allows the programming of the switching on of household appliances, giving higher priority to appliances that consume more energy. In addition to power consumption, the switching times, operating range, and frequency of use for each class of equipment are important parameters for load management. These parameters are provided by NILM systems. In [145],[146], the reduction in the cost of electricity is achieved by formulating the problem as a minimization problem, taking into account the scheduling flexibility of household appliances by consumers. The results show that the system can reduce the cost of energy for consumers in a meaningful way. These load scheduling techniques facilitate the time shift of existing loads in off-peak periods, so costs are minimized.

Çimen et al. [147] propose an NILM-based EMS to manage the operation of household appliances, in coordination with renewable sources and batteries within a residential microgrid. Residential grids include energy storage systems, some distributed generation units and one or more homes. These electrical systems can be used both in connected mode and in stand-alone mode, thanks to their flexible structure. Fig 3.2 shows the structure of the proposed microgrid.

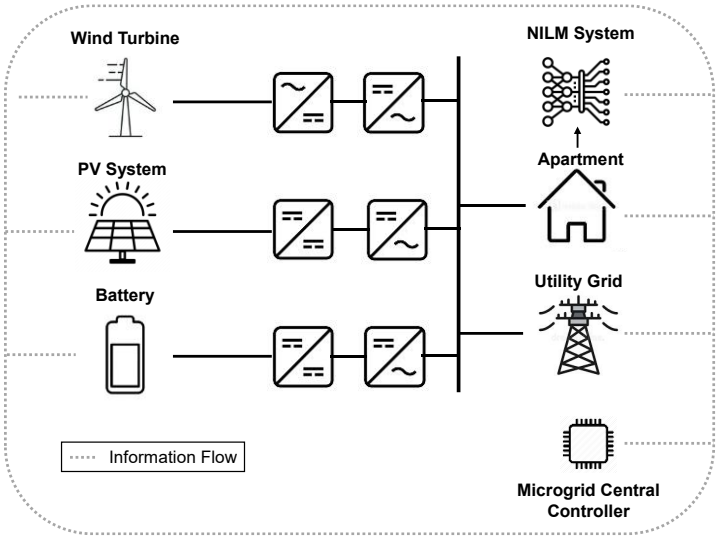

Figure 3.2: The microgrid structure with generation and storage sections.

On the power supply side, the energy sources include the public grid, a photovoltaic system, a wind turbine, and an energy storage system unit. Each system is managed by a microcontroller, through the information obtained from a NILM system, such as appliance activations, turn-on and turn-off times, power absorbed in the various operating conditions, and periods preferred by the user. The system optimizes the energy supplied by renewable source plants by minimizing the absorption of electricity from the grid.

In [148], Xia et al. propose an EMS that allows minimization of the cost of domestic electricity while satisfying the needs of comfort and safety. The system consists of a first module called the Solar Energy Management System which maximizes solar energy utilization, and a second module called the Appliances Scheduling System which minimizes the electricity load during peak hours. Using this EMS, the running time of the appliance is automatically programmed based on considerations concerning the preferences of the family, the day-ahead electricity price, and the historical data of electricity use. At the same time, the system uses the photovoltaic system and the storage systems in order to consume the minimum amount of energy from the grid. In the two subsystems that make up the EMS, a smart meter is used to collect the reading of the household's overall energy consumption and transmit it to the energy disaggregation module.

The framework modules, shown in Fig. 3.3, are described below.

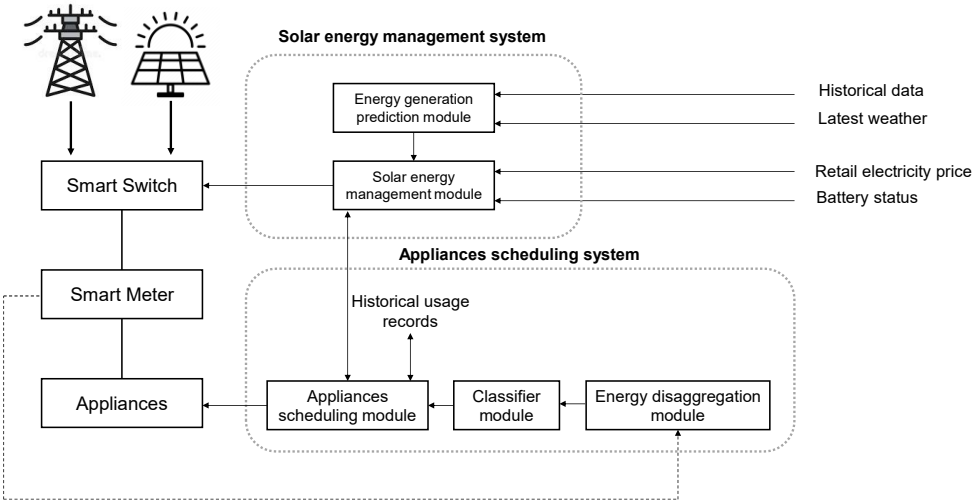

Figure 3.3: The EMS framework which illustrates the smart switch management system and the necessary modules.

- *Energy generation prediction module*: this module generates a forecast based on historical meteorological data for the considered position.
- *Solar energy management module*: this module controls the switch starting from the output of the previous block, the state of the battery, and the price of electricity. The various loads can be powered by both the solar system and the electricity grid; therefore, the system will include a controlled switch capable of connecting both power sources by absorbing energy from one or the other source. Therefore, this module optimizes the use of the collected solar energy. When the photovoltaic energy production exceeds the load demand, the photovoltaic energy is used as a priority, and the surplus energy is stored in the storage system. Only in the event that the photovoltaic energy (generated during the day or stored during the night) is not sufficient to support all the appliances in operation, does the system absorb energy from the grid.
- *Energy disaggregation module*: in order to implement efficient programming, the EMS should be aware of the details of the runtime of each device to deduce the activities of users on an average day. The preferences and habits of users are deduced on the basis of this detailed information at the appliance-level by imposing a certain level of priority, so as not to upset their routine too much. Therefore, to obtain appropriate planning strategies for a certain household, the consumption patterns of each appliance are key information.
- *Classifier module*: this module is designed to discriminate between the programmable and non-programmable devices. The classification

process is based on various parameters such as the flexibility and user preferences for each device. Initially, the classification module generates a profile of user preferences, preset according to the varied devices and historical data.

- *Appliances scheduling module*: this module classifies appliances as programmable or non-programmable and is responsible for programming the start time of programmable ones.

3.2.2 NILM systems in EMS

The optimal scheduling of appliances is traced back to an optimization problem, formulated as (3.1), solved through dynamic programming algorithms to search for sequences of states of each appliance that lead to a lower cost of electricity at the end of the day:

$$\min\left(cost = \sum_{i=1}^{N} \sum_{t=t_1}^{t_n} c_{(t)} \cdot p_{m,i(t)} \cdot t \right) \tag{3.1}$$

where cost is the sum of the costs of the energy absorbed by the grid, *N* is the number of schedulable appliances $t \in \{t_1, \dots, t_n\}$ is a set of discrete time intervals over a whole day (configurable by each user), and c_t is the price of electricity over the interval *t*.

The problem appears to be a constrained optimization problem since the time intervals that can be taken into consideration in the search for the minimum must be compatible with the information on the user behavior of the appliances, obtained from energy disaggregation.

Each appliance is characterized by a working mode m which is associated with both a specific power absorption, p_m, and a different impact on the result. For example, a dishwasher that performs a wash in eco mode has a reduced energy consumption as it heats less water, but takes longer to finish the cycle, with a consequent impact on user comfort. Therefore, the optimization algorithm must find a working mode that minimizes the cost of the bill while maintaining certain levels of comfort.

To take into account the user's satisfaction, the parameter s is introduced, which is equal to one when the appliances work in the period preferred by the user and decreases as this period varies. Its distribution model can be changed by a factor ξ,

depending on the degree of tolerance. Fig. 3.4 shows the distribution of user satisfaction as a function of a hypothetical appliance operating time.

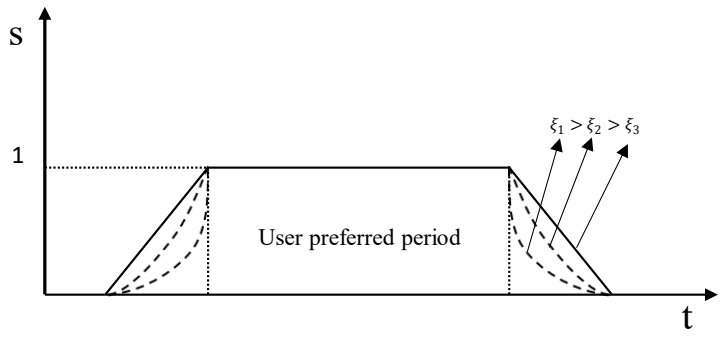

Figure 3.4: User satisfaction level and degree of tolerance.

The scheduling algorithm must therefore also try to maximize the user's comfort level. This problem can be formulated as in (3.2):

$$\max\left(comfort = \sum_{t=t_1}^{t_n}\sum_{i=1}^{N} s_{i(t)}\right) \tag{3.2}$$

where $s_{i(t)}$ corresponds to the user satisfaction value, introduced above, at time t for appliance i.

As previously introduced, the parameters necessary for scheduling household appliances can be obtained with the aid of the NILM algorithm. As an example, Fig. 3.5 shows the power distributions of two appliances over 24 h.

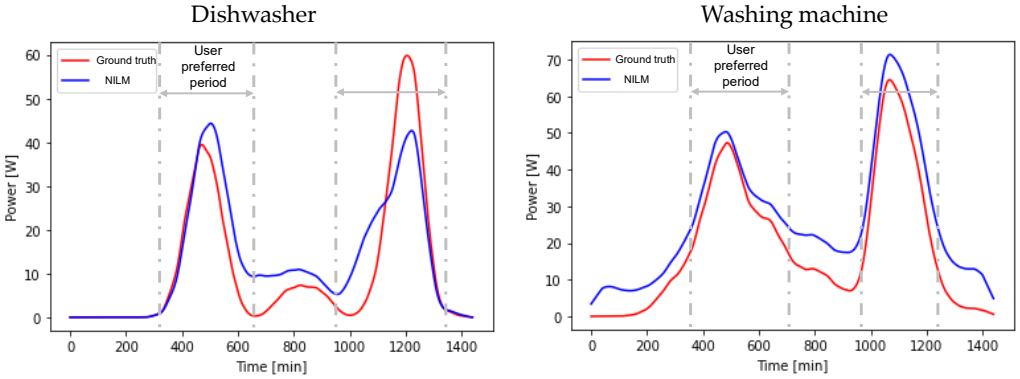

Figure 3.5: Power distribution of a dishwasher and a washing machine.

We obtained these distributions by processing the power measurements available from the REFIT dataset [149], which provides aggregate and appliance-level power metering for several homes over a period of approximately two years. The red curve rep-resents the power distribution, calculated with the power measurements at the appliance level. We obtained the blue curve by processing the corresponding aggregate power signal through an artificial Sequence-to-Point neural network, implemented in accordance with [113]. Since the REFIT dataset contains aggregated power and power supply data at the appliance level for 21 homes, data from homes that had washing machines and dishwashers were used for training the model. A house not seen during the training phase was therefore chosen to obtain the analysis shown in Fig. 3.5. The peaks indicate the periods when the device is used most frequently. Therefore, if the appliance is scheduled outside the gray dotted lines, customer satisfaction could be reduced according to the curve in Fig. 3.4.

Similarly, in [150] the authors propose a NILM system based not on DL as in [147], but on Bayesian theory. The proposed system also provides for the scheduling of appliances in order to optimize the power flows in a Smart Home. In [151] the IoT criteria for implementing an EMS based on NILM are outlined. In particular, the cited work refers to FHMM algorithms, but what was said can also be extended to other approaches such as DL, Combinatorial Optimization, and event-based.

An NILM system serving an EMS must have a high level of disaggregation accuracy; therefore, it must be able to determine, as closely as possible, the energy consumption profiles at the appliance level. It must also be able to provide information on the status (ON or OFF) in which the devices are positioned in order to guarantee correct feedback in scheduling. Comparing EMSs based on NILM with those based on ILM, it is clear that the main drawback linked to the use of NILM systems lies in their lower accuracy in estimating information, as can also be guessed from Fig. 3.5. In fact, NILM algorithms will always present greater uncertainty in estimating consumption and appliance status compared to intrusive systems. However, the use of NILM systems allows one to obtain the necessary feedback for energy management through a single installation, which represents a significant advantage, especially for those homes that were not born “smart”.

3.3 Demand response in smart grids

The spread of energy production plants from non-programmable renewable sources, such as wind and sun, requires greater flexibility of the electricity system, to ensure grid stability. With energy being difficult to store on a large scale, the balance between supply and demand can sometimes be delicate. To meet these needs, active demand management programs are created, ensuring greater flexibility and efficiency in the energy infrastructure.

Demand Response (DR) refers to the active management of energy demand, which involves the modulation of energy demand by commercial and industrial consumers as market conditions change. For example, in Italy this method was introduced by the Regulatory Authority for Energy, Networks and the Environment (ARERA) and is used by the operator of the national electricity transmission grid (Terna). DR mechanisms allow commercial and industrial consumers to respond to market signals by increasing or reducing their energy consumption, with the aim of responding to peaks in the de-mand or supply of electricity. This allows for greater flexibility and stability of the grid, and more efficient use of infrastructure and energy resources.

As an example of the functioning of the DR, we can consider the following steps.

- The network operator foresees a network stability problem and sends a balancing request to the aggregator (i.e., the intermediary between the dispatcher and the end users).
- The aggregator notifies its customers of the dispatching.
- The customer manually or automatically implements the modulation plan to reduce the load.
- The customer receives remuneration in exchange for the dispatch capacity.

The DR solution makes it possible to generate revenue for consumers based on their flexibility. With DR, grid operators can reduce energy consumption at peak times and consequently, the production of additional energy, improving grid stability. The two forms of DR already consolidated are:

- DR associated with network interuption tolerance: the consumer voluntarily chooses to reduce their withdrawals from the grid without notice via disconnection in exchange for an economic consideration (the minimum power is 1 MW).
- DR linked to multi-hour tariffs: prices change according to the band and day of withdrawal, thus stimulating users to consume energy in moments of lower load and less grid congestion.

- There is also a third innovative form of DR: The Market for Dispatching Services (MDS), a tool through which the national operator procures the resources necessary to manage and control the system.

In addition to large production plants or industrial loads, even users who have residential energy storage systems can offer balancing services to the electricity grid. It is therefore also possible for residential users to actively contribute with their energy storage system, rather than passively with simple connection/disconnection. The end user can choose, based on the opportunities present in the MDS, whether to withdraw or sell, store, or consume energy. To manage the storage system, the aggregator installs a communication and remote-control system.

3.3.1 The role of NILM in DR programs

A further step towards an increasingly distributed and sustainable model can be obtained by integrating NILM systems into DR programs aimed at residential users. As already discussed for the EMSs, NILM systems allow the aggregator to know the flexibility parameters of a user, making it possible to formulate an advantageous offer that does not affect the comfort of families.

Lucas et al. [152] proposed an algorithm for estimating the flexibility of the different electrical loads for DR purposes. The interactions between the actors are shown in Fig. 3.6. The request for a DR in a certain time window is reported by the network operator to the aggregator or to the Balance Service Provider (BSP). The BSP evaluates availability and flexibility, on the basis of user data, made available by an NILM system. The BSP interacts with the network operator and subsequently with the users. In particular, the NILM system provides information about the flexibility of users by analyzing the timestamps relating to the operating intervals.

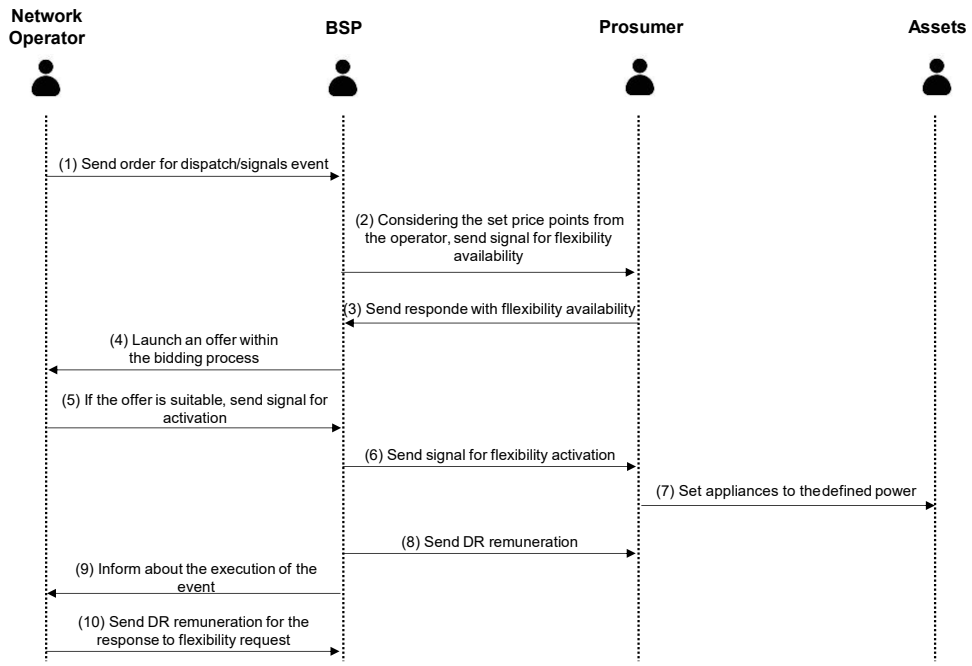

Figure 3.6: Illustration of the interaction between the actors in a DR sequence.

Fig. 3.7 shows an example of the information supplied by an NILM system, obtained in a similar way to that described for Fig. 3.5. The figure illustrates a case relating to the operating cycle of a dishwasher over a 24 h window. It should be noted that the greater length of the operating cycle, indicated by the actual measurement, is characterized by very low power consumption, which is therefore negligible. In fact, the NILM system in this particular case demonstrates an excellent generalization capability.

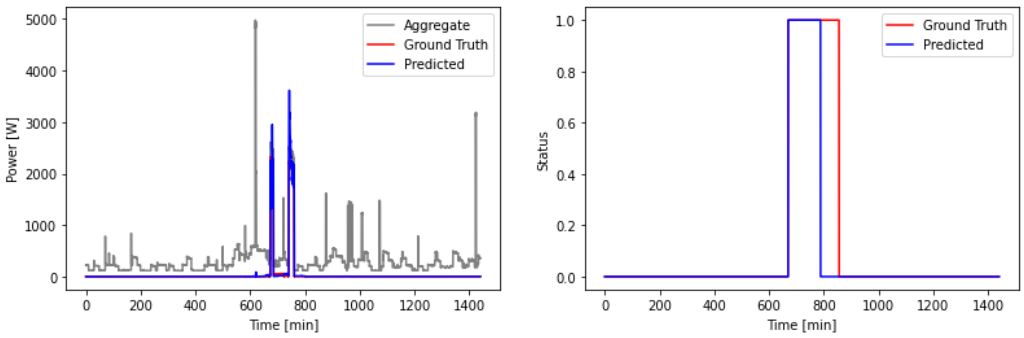

Figure 3.7: Example prediction of the user's flexibility in using the dishwasher, obtained by processing data related to a house in the REFIT dataset not seen during the training phase.

The applicability of NILM systems within smart grids was evaluated in [153]. In particular, the following were evaluated: the opportunity of deferred use of

electrical loads during the handling of consumption peaks; the possibility of proposing discount deals or time-of-use pricing programs to incentivize customers to postpone part of their energy demand; and finally, the potential of customers to defer loads.

This study was also conducted using the REFIT dataset [149]. Since the goal is to simulate a small smart grid, they assumed that the REFIT houses were connected to the same sub-grid, focusing the analyses over a three-month period (April to June 2014). The houses were chosen because, being in the same region of England, they are subject to the same climatic conditions. Therefore, by adding the profiles of total power absorbed by them, it was possible to simulate a smart grid.

In analyzing the total energy consumption in the different days of the week, some interesting trends were noted. Weekends have a higher average energy consumption than weekdays (which is predictable, as occupants leave for work and/or school reasons). The energy consumption profile presents peaks during a 5 h block in the evening from approximately 16:00 to 21:00 and similarly during a smaller 3 h block in the morning from approximately 5:00 to 8:00 (in line with typical school/working hours). Two energy peaks also occur on weekends, with an evening peak from around 16:00 to 21:00 and a peak in the morning from around 7:00 to 11:00. However, although it is lower than in the morning and evening, energy consumption even remains high during the day.

Based on these models, peak and off-peak hours were defined. Peak periods of consumption can increase the risk for the grid, so the goal is to find opportunities to level demand by encouraging customers to change their behavior. The loads that have the greatest advantage in terms of deferral were identified by carefully examining the energy consumption of the various houses and appliances during peaks in energy consumption at the grid level. As previously discussed, appliances such as washing machines, dryers and dishwashers are potentially deferrable appliances, as they do not have a great impact on comfort (i.e., they do not affect lifestyle by changing mealtimes or free time), so attempts are made to postpone these loads at off-peak hours.

Table 3.1 shows a summary of the power consumption including the total and deferrable loads (dishwasher, washing machine, dryer, etc.) for each home, during peak and off-peak periods, and the consequent potential cost savings. In off-peak periods, no distinction is made between deferrable and non-deferrable loads as there is no need to move them. The analysis in Table 3.1 assumes that loads such as dishwashers, washers and dryers are disaggregated through an NILM system, which produces estimates subject to accuracy limits. To this aim, they conducted further analysis on the influence of the accuracy of NILM disaggregation on the total loss

of revenue in a variable rate DR program. At the design stage, it is essential to understand how disaggregation inaccuracies affect the estimation of results in DR programs, as this allows decisions to be made that take into account the opportune margin of uncertainty. It is also essential to be able to rely on extremely efficient NILM systems, which allow one to obtain results that are as truthful as possible.

Table 3.1: Potential savings achievable with load deferral.

	Peak Periods			Off-Peak Periods	% of total load during peak periods	% of total load during peak periods that is deferrable	% of total load during peak periods if all deferrable load is moved to off-peak	Total energy cost under flat tariff ($)	Total energy cost under variable tariff without load deferral ($)	Total energy cost under variable tariff with load deferral ($)	Potenital savings from switching tariffs ($)	Likely to switch tariffs and change behavior?	Total energy cost assuming cheaper option is selected ($)
House	Deferrable load (kWh)	Non-deferrable load (kWh)	Total load (kWh)	Total load (kWh)									
1	11.6	296.9	308.5	376.2	45.1%	1.7%	43.4%	136.9	126.8	123.3	13.6	yes	123.3
2	92.4	370.1	462.5	304.7	60.3%	12.0%	48.2%	153.4	177.1	149.4	4.1	yes	149.4
4	23.8	305.4	329.3	381.2	46.3%	3.4%	43.0%	142.1	134.3	127.2	14.9	yes	127.2
5	189.4	587.3	776.6	562.2	58.0%	14.1%	43.9%	267.8	299.9	243.1	24.6	yes	243.1
6	14.2	412.0	426.2	503.7	45.8%	1.5%	44.3%	186.0	174.3	170.1	15.9	yes	170.1
7	171.0	350.4	521.4	412.4	55.8%	18.3%	37.5%	186.8	203.1	151.8	35.0	yes	151.8
8	34.8	444.6	479.5	752.5	38.9%	2.8%	36.1%	246.4	205.4	195.0	51.4	yes	195.0
10	112.9	606.2	719.1	624.0	53.5%	8,4%	45.1%	268.6	282.9	249.0	19.6	yes	249.0
12	33.1	345.5	378.6	279.1	57.6%	5.0%	52.5%	131.5	146.5	136.5	-5.0	no	131.5
13	83.7	564.3	648.0	395.8	62.1%	8.0%	54.1%	208.8	246.6	221.5	-12.7	no	208.8
15	43.4	234.8	278.2	288.5	49.1%	7.7%	41.4%	113.3	111.8	98.8	14.6	yes	98.8
16	81.0	533.3	614.4	558.3	52.4%	6.9%	45.5%	234.5	242.9	218.6	15.9	yes	218.6
17	8.9	300.8	309.7	298.0	51.0%	1.5%	49.5%	121.5	123.3	120.6	0.9	yes	120.6
18	36.4	389.4	425.8	463.6	47.9%	4.1%	43.8%	177.9	172.2	161.3	16.6	yes	161.3
19	3.4	260.6	264.1	270.1	49.4%	0.6%	48.8%	106.8	105.9	104.9	1.9	yes	104.9
20	23.7	309.8	333.5	325.8	50.6%	3.6%	47.0%	131.9	133.0	125.9	6.0	yes	125.9
Total	963.7	6311.4	7275.4	6796.1				2814.2	2886.0	2597.0	217.3		2579.3

3.4 Anomaly detection and maintenance

As previously discussed, NILM systems continuously monitor the absorption of individual devices connected to the electrical grid. An innovative application is to use these systems to detect anomalies in electrical loads. To accomplish this task, the NILM systems, in addition to estimating the energy consumption of the appliance, must also be able to faithfully determine the power absorption profile over time, which is then analyzed by the anomaly identification systems.

3.4.1 Anomaly detection with NILM

In the works [154]-[156], the first NILM systems for the detection of anomalies were proposed. These systems make it possible to identify faults in air conditioners, refrigeration systems, and waste disposal systems powered by autonomous circuits. Rashid et al. [157] presented a study on the ability of an energy disaggregation system to identify the anomalies of household appliances inside an apartment. The algorithm, based on a Factorial Hidden Markov Model (FHMM), starts from the measurement of the aggregate power supplied by the smart meter and is based on pre-established rules to estimate the type of anomaly and its temporal position.

The study focuses on the analysis of refrigerators and air conditioners, as common, high-consumption, compressor-based appliances. Any failure of the compressor, or of any other part that affects it, is reflected in the profile of absorbed power. Depending on the type of fault, the durations of the ON and OFF conditions differ significantly from the nominal ones, although these deviations could also be due to different causes.

For example, an elongated duty cycle may be due to a clogged air conditioner filter, its incorrect configuration or a failure of the refrigerator door. Similarly, a higher switching frequency between the ON and OFF states can occur, due to damage in the compressor, short circuits, or refrigerant leaks. Appliances such as refrigerators have different absorption profiles depending on the time of day, as shown in the Fig 3.8.

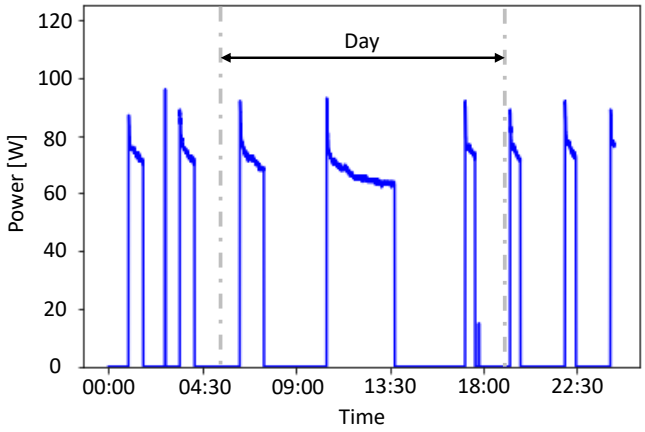

Figure 3.8: Power profile of the refrigerator during 24 h.

Furthermore, a malfunction can result in both higher and lower power consumption. Therefore, using the total daily consumption to estimate anomalies may not be accurate. As can be seen from Fig. 3.8, both anomalous conditions result in a daily energy consumption higher than that of normal behavior. An observation based solely on daily energy consumption does not explain the type of anomaly, i.e., if there was a problem with an elongated duty cycle or more frequent switching between the ON and OFF state. For these reasons, the authors have proposed an analysis of the average energy consumption taking into account the characteristics of the device; the method is called UNUM. UNUM is an anomaly detection algorithm consisting of a training and a test phase. During the training phase, the statistical model of the device is built, starting from the profile of the power absorbed during normal operation. During the test phase, the input is the profile of the power absorbed by the device during the day, and the algorithm issues indications on the presence or absence of anomalies.

The identification of the ON / OFF states of the compressor occurs through a clustering algorithm known as k-means [158], which provides two clusters corresponding to the states and the relative power measurements. Note that the power absorbed during the OFF state is not zero because a device can continue to absorb power even when the compressor is OFF.

The time of the power measurements at the extremes that define the single state are identified. From these measurements, we calculate the duration, Ds, and the energy consumed in each state, Es. For both the ON and OFF states, the means of the Ds and Es are calculated, associating them with $\overline{D}$ and $\overline{E}$, respectively. Furthermore, the standard deviation of Es, denoted by σ_e, is calculated. During the analysis, it was observed that the air conditioner and the refrigerator draw power differently at different times of the day, which is logical. Fig. 3.8 shows the

signature of the work cycle of the appliance during the day and night, and significant differences are noticed. Therefore, separate models were built: one for the day (06:00–18:00 h) and the other for the night.

The algorithm first takes the data on the power profile absorbed by a device and calculates all the parameters described above. Next, the method described below is used to decide if the profile has anomalies and, if so, what kind of anomalies they are (increased duty cycle or switching frequency):

a. If an appliance frequently passes from the ON to the OFF state and vice versa, as for "abnormal behavior 2" in Fig. 3.9, in an ON / OFF cycle, it will consume less energy than in the normal case (presenting, however, a higher consumption in the overall operating time). Therefore, an anomaly associated with "anomalous behavior 2" must be identified as follows:

$$E_{test}^{i} < \bar{E}^{i} - n \cdot \sigma_{e}^{i}, \quad \forall\, i \in \{ON, OFF\} \tag{3.3}$$

where $\{n \in R | n > 0\}$ is a control parameter that determines how many standard deviations from the historical profile should be tagged as an anomaly. Intuitively, *n* controls the granularity of the anomaly: a small value of n means that an anomaly is flagged if a minor deviation is observed and vice versa.

b. If an appliance remains in its ON state for an excessively long period (prolonged work cycle), the energy consumed within a cycle will be greater than normal:

$$E_{test}^{ON} > \bar{E}^{ON} + n \cdot \sigma_{e}^{ON} \tag{3.4}$$

c. If an appliance remains on for a prolonged period and remains off for a longer period than normal, this is not an anomaly, because it is a condition that occurs when the appliance is switched on after a long period:

$$E_{test}^{ON} > \bar{E}^{ON} + n \cdot \sigma_{e}^{ON} \quad \Lambda \quad D_{test}^{OFF} > \bar{D}^{OFF} + n \cdot \sigma(D) \tag{3.5}$$

This set of rules allows the UNUM algorithm to indicate the status of the anomaly in an informative way, so as to allow a quick decision-making process on which

type of anomaly is present and, since two separate models are used for day and night, in which part of the day the anomaly occurs.

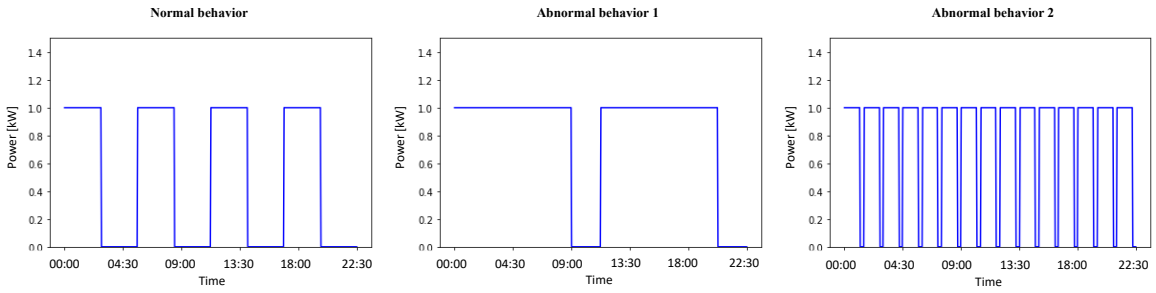

Figure 3.9: Hypothetical cases of elongated duty cycle (abnormal behavior 1) and increased switching frequency (abnormal behavior 2).

To evaluate the effectiveness of the UNUM algorithm, the data obtained by the NILM system were processed, comparing the results with those obtained by applying the algorithm to the data measured directly on the appliance.

The following metrics were used to indicate the accuracy of both the NILM and anomaly detection systems:

$$Appliance\ Normalized\ Error\ (ANE) = \frac{\sum_t |y_t^a - \hat{y}_t^a|}{\sum_t y_t^a} \tag{3.6}$$

where y_t^a represents the power measured at the appliance level for the appliance a and $\hat{y}_t^a$ represents the corresponding estimate provided by the NILM algorithm. The lower the ANE value, the more precise the disaggregation algorithm will be for the appliance a:

$$Precision = \frac{TP}{TP + FP} \tag{3.7}$$

$$Recall = \frac{TP}{TP + FN} \tag{3.8}$$

$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{3.9}$$

With precision, the percentage of true anomalies detected is taken into consideration and compared to the total number of anomalies detected (and

therefore alarms triggered). In addition, the percentage of true anomalies detected is maintained and compared to the total number of anomalies (detected or not) and compared to the recall. The higher the F1 score's value, the more accurate the anomaly detection algorithm will be.

The results showed that a good number of abnormal cases for an appliance can be correctly identified if the ANE is less than 0.1. The choice of the NILM technique is therefore determined by the ANE value achieved for the household appliances of interest.

It was observed that as control parameter n increases in the UNUM algorithm, the recall decreases. This is because, as n increases, observations with a lower standard deviation are treated as normal observations, thus increasing false negative results. At the same time, an increase in n makes it less likely that small deviations are considered anomalies, resulting in fewer false positives (and therefore false alarms) and a higher precision value. An accurate choice of the control parameter n based on the trade-off between recall and precision is therefore essential.

3.4.2 Condition-based maintenance

Condition-Based Maintenance (CBM) refers to all those activities aimed at maintenance, which are based on the data collected by monitoring the status of the equipment. This type of maintenance differs from the traditional one, which instead is performed on a scheduled basis or when faults occur. Failure alerts are provided from the collected data to help plan equipment repairs or replacements.

The goal of a CBM system is to detect minor failures and take action before they turn into more dangerous failures. These minor faults are often invisible to operators and therefore require analysis of the electrical power measurement. An NILM system can allow CBM to be an effective tool by detecting and identifying both equipment signatures and their anomalous behaviors. An important feature of these systems is the presence of a user interface that helps the operator make operational decisions in an intuitive way. An NILM system was deployed aboard the United States Navy [159] to monitor energy and faults in electrical installations with minimal invasiveness, thereby installing a minimum number of sensors to monitor various loads on board. The loads monitored by the NILM Dashboard [160] are motors, engines, generators, and pumps. The monitored loads are all the ON/OFF type (Type I), with the exception of a diesel purifier, which is an FSM and for which each state is considered separately. The NILM systems installed on ships have two primary objectives: the first is to detect anomalies to improve the functioning of the system, and the second is to give operators better awareness by identifying the

operating programs of the equipment. The NILM system used is an event-based system, i.e., a first algorithm is used to detect events, and then an artificial neural network, trained to identify anomalous situations starting from the measurement of active and reactive power on the three phases, is used.

In this way, the NILM system detects all those events that occur as a result of status changes of the various loads, thus acquiring knowledge on the operating programs of the individual equipment.

More specifically, there are five parameters for diagnostics:

- the active and reactive power absorbed in stationary conditions;
- the power factor;
- the time between activation and shutdown, indicated as Average Run Duration;
- the total time in which the equipment is operational over 24 h, indicated as the Total Run Time;
- the number of discrete operations per day, indicated as Daily Actuations.

These metrics help detect degradation in equipment material, such as that which occurs from mechanical wear and corrosion. The wear of a motor bearing can be reflected as a change in power consumption [161], while corroded heaters could be reflected as a change in power factor. The NILM system monitors heaters and pumps which are controlled by automatic closed-loop systems based on thermostats or tank-level sensors. Therefore, the Daily Actuations, total run time and average run duration are also useful in identifying sensor and automation failures that could cause too frequent stimulation or insufficient operation. In fact, a failed thermostatic controller or a broken tank indicator can cause excessively long or repeated periods of equipment activation [162]. However, it is important to note that frequent pump activations or even a single long activation are not necessarily cause for alarm. This is taken into consideration by monitoring the 24 h average of the Average Run Duration, power, and power factor parameters, and taking into account the Daily Actuations and Total Run Time. The 24 h evaluation prevents false alarms following brief anomalies. Depending on whether the loads are activated less or more frequently, stricter controls are necessary, so the period can be easily adjusted for various applications.

In [163], techniques were presented to identify progressive failures starting from measurements obtained through a NILM system and provide analysis of vibrations, materials, and the analyzed electrical signatures. In the work presented in [155], performances obtained using FHMM are shown, and in [164] the authors propose

a similar system for the detection of anomalies on the consumption patterns of household appliances, but instead based on an autoencoder.

The CBM parameters described above are communicated on the NILM dashboard through diagnostic indicators divided into three levels of signaling, namely, healthy operation, fault warning, and definitive fault alarms, as shown in Fig. 3.10. Establishing the correct threshold for each region of the indicators is critical in order to be usable by the ship's crew. Several techniques were proposed to determine the failure thresholds related to industrial applications. In [165], the authors proposed a statistical process control (SPC) method, where an NILM system collects data that are used for SPC analysis. The deviation in parameters from historical data is an indication of a possible failure. SPC allows for the exact determination of when a deviation should determine a fault warning and how much deviation is acceptable. The system defines a center line, a lower control limit (LCL), and an upper control limit (UCL). If a parameter reaches the lower or upper control limits, warnings are issued. SPC analysis defines the center line as the arithmetic mean of the various parameters considering a standard normal distribution for each parameter. In this application, a normal distribution does not adequately fit the data. The best results are obtained using a Weibull distribution, typical of many machinery reliability applications [165].

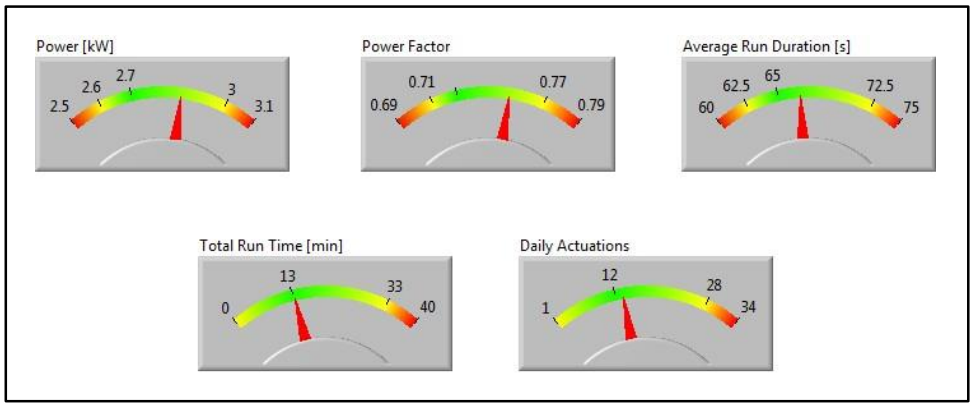

Figure 3.10: The NILM dashboard which reports the measured parameters and the related alarm thresholds.

An interesting consideration is that if an operator knows that some piece of equipment should be in operation, but the NILM-based CBM system shows no activity, i.e., the daily actuations have dropped to zero, there is evidence of a failure. This can happen for several reasons and is not necessarily indicative of equipment failure. First, it could indicate a complete failure of some piece of equipment. Second, it could be due to a broken sensor, as a result of which the load does not

turn on, even when it should. Alternatively, it could be due to a degraded part of the equipment, which causes a change in absorption compared to the value considered normal by the NILM, so the load is actually functioning but the NILM does not consider it to be so.

However, to help a user identify abnormal behavior, an NILM-based CBM system should track changes not only in the behavior of the equipment in terms of the number of starts and stops but also in the power draw of the device. The problem lies in the impracticability of training NILM systems on all anomalous signatures because the variations depend on the type of anomaly. Therefore, the NILM identification algorithms are trained on the data obtained during correct operations. Nevertheless, a priori knowledge of the operation of the equipment can help define the equipment even in its degraded state. CBM and electromechanical fault detection systems can be developed cost-effectively and quickly through the use of NILM systems. The load analysis discussed in [157] shows that through a selection of statistical parameters, these systems can analyze a wide range of system anomalies and assess the health of individual equipment.

3.5 Disaggregation of regional demand

Distributed energy resources (DER) refer to the decentralized production of electricity through small self-production units located in different sites and connected to the distribution network. This model is distinct from the traditional centralized model which relies on a small number of large power plants connected to the transmission grid. These resources, found both in remote locations such as wind farms and in proximity to end users like cogeneration plants, are generally connected to low voltage distribution networks.

DERs play a crucial role in the development of sustainable smart cities, but monitoring loads on a distributed network can be challenging for distribution network operators. Additionally, it is not practical to install sensors for every resource on the network. To address these problems, the NILM technique has been adapted to monitor DERs connected to regional substations, as demonstrated in [166]. Specifically, the system was used to disaggregate regional demand into traditional load, flexible load and distributed generation components, as shown in Figure 3.11.

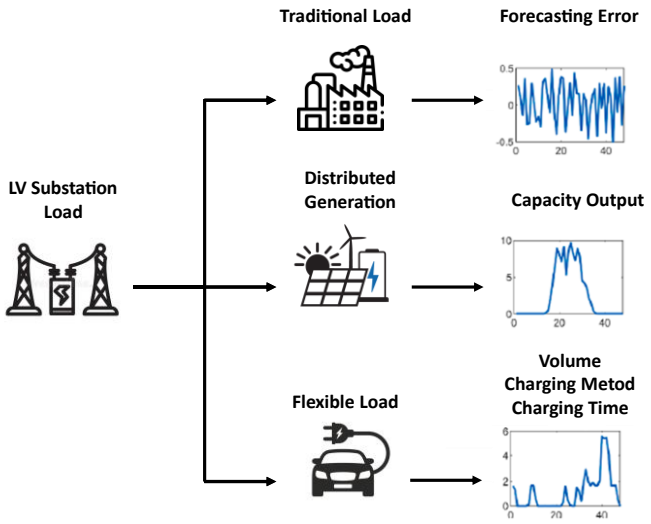

Figure 3.11: Energy disaggregation at LV substation level.

The electricity generated by distributed sources such as photovoltaics or small wind turbines is considered a negative load. As a result, the power obtained by adding the total load and the distributed generation allows for multiple disaggregation solutions, each made up of different combinations of distributed loads and generators. Therefore, using NILM systems to separate these two contributions can result in non-unique solutions. To address this issue, the authors

propose a three-step approach. The first step consists in predicting the traditional load, consisting of loads geographically connected at fixed points and temporally invariant in terms of quantity, using the latest generation NILM systems based on long-term historical data. The residual of this forecast is deemed a forecast error, linked to distributed generation and flexible load. In [166], flexible load refers to electric vehicles connected to a substation which vary in time due to mobility. Additionally, their charging and discharging patterns can vary widely depending on the brand and charging mode.

This approach can be formulated as in (3.10).

$$P_{sub} = P_{trd} + P_{DER} + P_{EV} \tag{3.10}$$

where P_{sub} is the output power from the regional substation, P_{trd} is the power relative to the traditional load, P_{DER} is the power supplied by the distributed energy resources and P_{EV} is the power absorbed by the electric vehicles.

The P_{trd} is derived using a NILM system trained on historical data. The error in the estimate can be defined as in (3.11).

$$Err_{trd} = P_{trd} - \hat{P}_{trd} \tag{3.11}$$

Consequently, the difference between the output power from the substation and the power estimated by the NILM system is:

$$\begin{aligned} \Delta\mathrm{P} = P_{sub} - \hat{P}_{trd} &= (P_{trd} + P_{DER} + P_{EV}) - (P_{trd} - Err_{trd}) \\ &= P_{DER} + P_{EV} + Err_{trd} \end{aligned} \tag{3.12}$$

Err_{trd} is almost a white noise while P_{DER} and P_{EV} show a strong periodicity (daily for P_{DER} and from 1 to 7 days for P_{EV}), therefore a spectral analysis was adopted in order to reduce the high frequency components of ΔP. In particular, empirical mode decomposition (EMD) [167] was used, which allows to decompose non-stationary data sets into a finite and often small number of components. These components are described as intrinsic mode functions (IMFs) and form a complete and nearly orthogonal basis for the original signal.

In this process, a total of n IMFs are obtained with a final residue r_n. The filtered residual value $\Delta\mathrm{P}^*$ can then be rewritten as in (3.13).

$$\Delta\mathrm{P}^* = \sum_{i=1}^{n} IMF_i + r_n \tag{3.13}$$

Frequencies decrease as n increases. Since the first u IMFs usually carry the most oscillating (high frequency) components, the error Err_{trd} is mainly distributed in this part. The specific value of u is determined by the frequency components of the IMFs. In general, the high frequency components are mainly concentrated in the first IMF. Reconstructing the remaining low frequency IMF and the residual r_n we obtain the (3.14).

$$P_{DER} + P_{EV} = \sum_{i=u+1}^{n} IMF_i + r_n \tag{3.14}$$

The second step consists in deducing P_{DER}, that is the contribution of the distributed energy resources. In the study taken as an example, the case in which the production of energy from distributed generators is exclusively photovoltaic (PV) was considered. Therefore, a widely used formula for estimating PV production is given in (3.15).

$$P_{PV} = A \cdot R \cdot PR \cdot \eta \tag{3.15}$$

where P_{PV} indicates the power produced (kW), A indicates the total area of the panels (m^2), η is the efficiency of the solar panels, R is the incident radiation (kW/ m^2) and PR is a performance ratio (or coefficient loss, between 0.5 and 0.9, usually set at 0.75, as R and η do not take into account shading). It is assumed that in a short period of time (the study in question considers 10 days) the installed area is a constant and the other coefficients are relatively stable. Downstream of these assumptions, we can write the (3.16).

$$P_{DER} = P_{PV} = K \cdot R \tag{3.16}$$

A peak matching analysis was proposed to estimate the value of K for the area and period studied. ΔP^* consists of only a mixture of EV load and PV generation. The idea is to look for a period in which a local minimum of the negative part of ΔP^*, which will be denoted by $\overline{\Delta P^*}$, matches with the local maximum of the incident radiation R, known on the base of the meteorological data being the meteorological conditions similar in the whole studied region. It is assumed that if the local maximum of solar irradiation matches with the local negative minimum of $\overline{\Delta P^*}$, the PV generation is predominant in the period considered. The estimated value of the K coefficient can be described as:

$$\widehat{K} = \frac{1}{Z_{\alpha=1}} \sum_{i=1}^{p} \sum_{j=1}^{q} \alpha \frac{(\overline{\Delta \mathrm{P}^*})_i^{idxi}}{(R)_j^{idxj}} \qquad \alpha = \begin{cases} 1 & |idxi - idxj| < \varepsilon \\ 0 & |idxi - idxj| > \varepsilon \end{cases} \tag{3.17}$$

where $idxi$ is the index of the i-th local minimum of $\overline{\Delta \mathrm{P}^*}$ and $idxj$ is the index of the j-th local maximum of R. Obviously $idxi,\ idxj \in [1, n]$ where n is the length of the signal considered. p and q are the number of peaks respectively of $\overline{\Delta \mathrm{P}^*}$ and of R. $(\overline{\Delta \mathrm{P}^*})_i^{idxi}$ is the value of the i-th peak of $\overline{\Delta \mathrm{P}^*}$ e $(R)_j^{idxj}$ is the value of the j-th peak of R. α is the match indicator, its value is equal to 1 when two peaks match, otherwise it is equal to 0. When the difference between the two indices $idxi$ and $idxj$ is less than a certain threshold ε, then the two peaks will be considered corresponding. $Z_{\alpha=1}$ is the number of matched peaks.

Consequently, we have that:

$$P_{EV} = \Delta \mathrm{P}^* - \widehat{K} R \tag{3.18}$$

At this point the output power measured from the substation has been disaggregated into the three contributions described above. In [166], the P_{EV} is further processed in order to determine the number of electric vehicles and the charging modes (slow, fast and extra fast charging) through limited activation matching pursuit (LAMP) [168].

The method described, called Regional-NILM, provides a way to estimate the real-time status of DERs without costly monitoring and data privacy issues.

Fig. 3.12 summarizes the entire procedure in three steps:

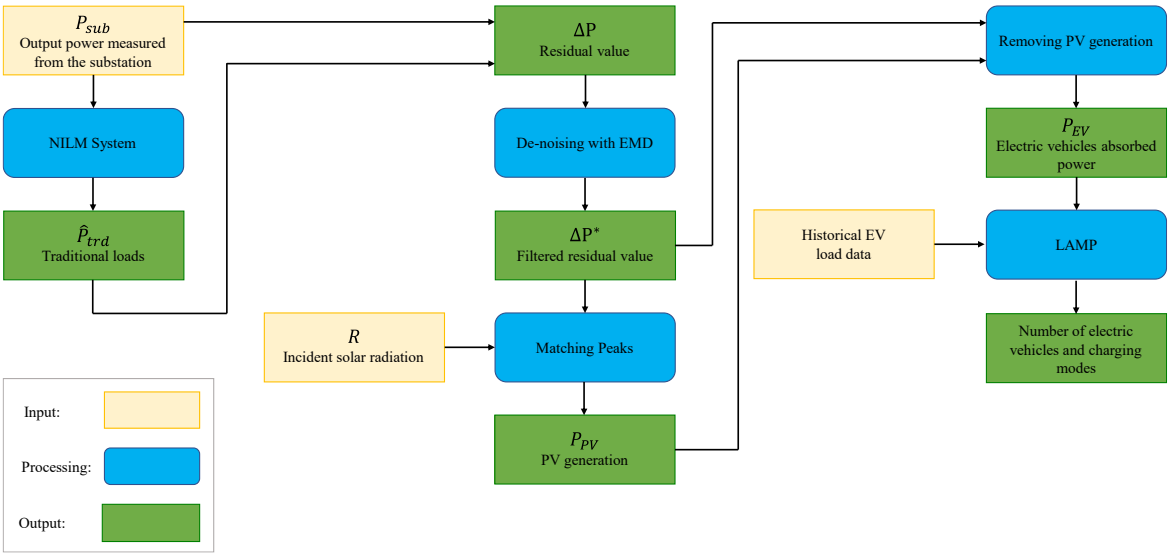

Figure 3.12: The flowchart of a regional demand disaggregation system.

Future research could investigate the use of deep neural network to manage the entire process, through multi-input structures able to process the output measured power from the substation and the incident solar radiation and directly provide the three power contributions.

3.6 Ambient assisted living

The concept of Ambient Assisted Living (AAL) represents a set of technological solutions (including home automation systems) designed to make the home environment active, smart, and collaborative. It must be effective in supporting independent living and be able to provide greater safety and well-being in carrying out the Activities of Daily Livings (ADL). These solutions go beyond the technological content, but also include aspects of design and the analysis of the needs of society, psychology, and medicine. This differentiates AAL from home automation, which is specifically focused on technology and automation in the home environment. Despite this, both technologies have something in common. Almost all AAL projects are focused on home automation. Knowing a person's ability to undertake normal ADL is an essential part of the AAL, as it allows for the diagnosis to be made. There are several ADLs that can be deduced from the data obtained with smart meters or sensors installed in the home, using NILM algorithms [169]. The following list highlights the main ADL that can be detected through a patient's interaction with their electrical devices:

- Eating patterns: for the purpose of detecting abnormal or altering changes in eating habits. These types of behavioral changes provide key indicators regarding the patient's overall health.
- Sleep Patterns: changes in sleep patterns can provide insight into a patient's mental and physical well-being. Sleep disturbances are often key indicators of various mental health problems.
- Routine alterations: it is vital for detecting changes in a patient's behavior and forms a fundamental part of the AAL system, in order to facilitate an independent life. Identifying a change of course, especially in more serious conditions such as dementia, may indicate the need for immediate intervention.

Nory et al. [170] proposed an NILM system that monitors the ADL starting from the identification of active electrical devices within the home. The algorithm identifies the various powered devices, recording information on their switching ON or OFF, from which it obtains useful information for the AAL purposes. For example, detecting a light turning on indicates that someone is entering a room. Consequently, every device in the house is virtually transformed into an additional sensor, without disturbing the user's privacy.

Specifically, the authors analyzed four activities (preparing and consuming food, hygiene, dressing, and grooming) in four periods (i.e., morning, day, evening, and night) and two additional activity levels (day and night).

Table 3.2 shows the relationships between rooms, appliances, and ADL. However, the relationship between individual appliance activation and ADL is not inductive. To this aim, a factor p_{ij} was defined that determines the weight of an appliance i in ADL_j. If the appliance is very representative of an activity, this will have a high weight for the classification of the activity; therefore, this weighing process allows one to take into account the functionality of each appliance. For example, the coffee machine, which is used frequently in the morning, receives the maximum weight in the "breakfast" ADL. On the contrary, the kitchen light receives the minimum (not zero) because there is a lower ratio between the "feeding" activity and the kitchen light. In the event that a room is only equipped with lighting, the room lights will have the higher weight in detecting this activity. As an example, for "hygiene" activity detection, the bathroom light will be the main indicator, unless another appliance such as a hair dryer is currently being used. Downstream of this, if at a certain moment a certain subset of active devices is detected, the activity carried out at that moment will be the one that maximizes the weight factors, in accordance with (3.19):

$$ADL(t) = \arg\max_{ADL_j} \left(\sum_{i=1}^{N} p_{ij} \cdot r_i(t) \right) \tag{3.19}$$

where N is the number of appliances and r_i is equal to 1 if appliance i has occurred, otherwise it is 0.

Table 3.2: Relationships between rooms, appliances, and ADL.

Room	Appliance	ADL
Kitchen	Kitchen light, fridge, furnace, boiler, dishwasher	Cooking, eating
Toilets	Toilet light, heater	Continence
Bathroom	Bathroom light, hairdryer, heater	Grooming
Other room	Light of the room	Other

Zhang et al. [171] used a Latent Dirichlet Allocation (LDA) algorithm [172] to create a model of the use of household appliances that allowed them to deduce the activities carried out by the occupants, starting from their absorption profiles. It is a probabilistic model used mainly for text processing, which deduces the semantic meaning of a document by analyzing the similarity between the distribution of the terms of the document with that of a specific topic. In a similar way, given a certain time interval, the states in which the appliances are found in that time interval are used to infer the activity (or activities) in progress. A system of this type can be

created using the information of an NILM system, allowing the costs and intrusiveness of the sensors to be minimized.

An AAL system makes the home smart by digitizing it through smart sensors and appliances that form a network capable of providing automated services to the user based on their lifestyle. To provide these services, the house equipped with an AAL system can use information from NILM systems, such as those described above, to primarily monitor the resident's activities. A general scheme of the system is shown in Fig. 3.13. The system is therefore able to analyze the data collected on the activities of the resident in his environment. Based on this analysis, the digital environment can offer tailor-made services for the resident and assist them in their daily life. Data from the NILM-based AAL system are transmitted over Wi-Fi to the homes of family members, friends, healthcare assistants, and doctors. In this way, it is possible to constantly monitor the health of the resident. In the event of an accident or emergency, rapid assistance can be provided through direct communication between the participants. It is evident that an NILM system designed for AAL applications must be able to determine the state (ON or OFF) of the various monitored devices. In this case, it is no longer necessary to know the absorbed electrical quantities. For example, in [173],[174], systems are proposed that are able to infer the ADLs starting from the power profile of each appliance, identifying normal and anomalous patterns. This type of approach is limiting compared to the systems described previously as it would not allow reactive interventions in a short time. Therefore, an easy-to-install system that allows the status of the various electrical loads to be analyzed is ideal for an application of this type [2].

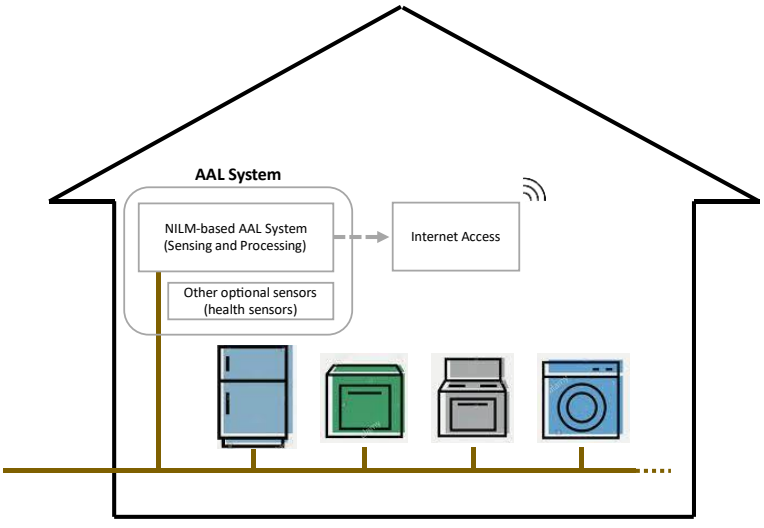

Figure 3.13: General scheme of systems using NILM in AAL, which makes it possible to identify the ADLs within a monitored apartment, often also thanks to the use of auxiliary sensors.

Chapter 4

Design, implementation and metrological characterization of innovative NILM systems

During the course of the Ph.D., extensive research activities were undertaken, leading to the development of three distinct NILM systems. Chapter 2 of the thesis extensively discusses NILM systems and their two macro-categories: event-based and non-event-based systems. In order to conduct a comprehensive study, three systems were implemented, each with different characteristics and capabilities.

- The first proposed system, discussed in Chapter 5, is an event-based system that samples the absorbed current signal at a high frequency of 10 kHz. The proposed system differs from most algorithms in that it detects the event and classifies the device through a single process, whereas most algorithms require a separate device identification phase. Moreover, this system is unique in its ability to recognize appliances from the measurement of the absorbed current alone. This feature allows the added advantage of making galvanically isolated measurement systems that are easy to install through CTs.

- The second system, discussed in Chapter 6, is a non-event-based system that processes an active power signal available at a significantly lower frequency of 1/8 Hz. Despite the lower frequency, this system has a high generalization capability that allows it to be installed on an electrical system without prior knowledge of the loads it will be

monitoring. The system, implemented on a single-chip microcontroller based on the ARM (Advanced RISC Machine) architecture, has been metrologically characterized by a long 12-month test phase. It updates the energy disaggregation data every 8 seconds and provides detailed information about the energy consumed by individual loads, as well as their status.

- The third proposed system, discussed in Chapter 7, is particularly innovative and departs from the definitions of NILM systems used in the literature. This system is based on the injection of a variable-frequency signal into a generic electrical socket of the system being monitored. The identification of electrical loads connected to the electrical system is done by processing the obtained frequency response using ML algorithms. Therefore, it does not rely on processing time sequences of absorptions in order to detect changes (as in event-based systems) or consumption patterns (as in non-event-based systems). This approach overcomes the typical problems of NILM systems in identifying multistate or continuous variable load appliances. Additionally, the system presents a simplified connection mode compared to traditional NILM systems, as it can be installed directly on an electrical socket.

All the three proposed systems underwent an extensive testing phase, where their performance in real-world scenarios was evaluated. The systems differed in the type of approach, the different sampling frequencies used for signal acquisition, the characteristics of the hardware required, and the type of contribution they can make to the context in which they are applied. These systems represent significant advancements in the field of NILM and can be applied in the various contexts discussed in Chapter 3.

Chapter 5

A high sampling rate event-based NILM system

The first NILM system that has been implemented is a system designed to be installed inside a home. This system is capable of acquiring and processing the overall user current. The proposed solution is a DL-based NILM system. It adopts a CNN, a particular type of ANN [175],[176], as previously explained in the Chapter 2.

A salient aspect of CNNs is that they are suitable for processing complex input data, such as multidimensional arrays. In the proposed application, the CNN processes the Short-Time Fourier Transform (STFT) of the total current. Although in most algorithms an event detection step is followed by device identification, in this work, event detection and classification of the related device are performed by the same and unique process.

The operational characteristics of the proposed system are verified by extensive measurements. The results obtained from field applications are also included and discussed.

5.1 Analyzed load signature features

Although NILM has been investigated for over two decades, no systematic selection of the electrical characteristics that allow for unequivocal load discrimination has been presented yet. Therefore, the identification of the most significant set of electrical parameters that allow them to be distinguished remains one of the biggest challenges.

The load characteristics, as already explained in Chapter 2, can be classified into stationary and transient-state characteristics based on the state of the measured waveform they represent. The load signature proposed in this chapter is based on the transient characteristics. More specifically, the transient characteristics are represented by the spectrogram of the derived rms current signal. By deriving the rms current, the steady states are filtered and all transient information is maintained. In this way, it is possible to classify an event, regardless of the load conditions present.

The load signature allows the action of a device to be identified when it comes into operation by measuring only the overall current of the monitored system. First, the current effective (rms) value is calculated by processing the acquired raw current with a sliding window technique, as follows:

$$I_{\mathrm{rms}\,(k)} = \sqrt{\frac{1}{N} \sum_{n=k}^{k+(N-1)} i_{(n)}^{2}} \tag{5.1}$$

where k is the kth measured current sample, N is the number of samples per cycle, $i_{(n)}$ is the sampled signal, and n is the summation index.

At the next step the derivative of this signal is calculated, obtaining the signal $I'_{rms\,(n)}$ shown in Fig. 5.1, whose pulses represent the transient states of the rms current. The location of the pulse in the derived signal identifies the instant in which a certain event occurred.

This impulsive signal is successively processed by the STFT through the following known standard transformation [177], [178]:

$$STFT_{(m,\omega)} = \sum_{n=-\infty}^{\infty} I'_{rms(n)}\, w_{(n-m)}\, e^{-j\omega n} \tag{5.2}$$

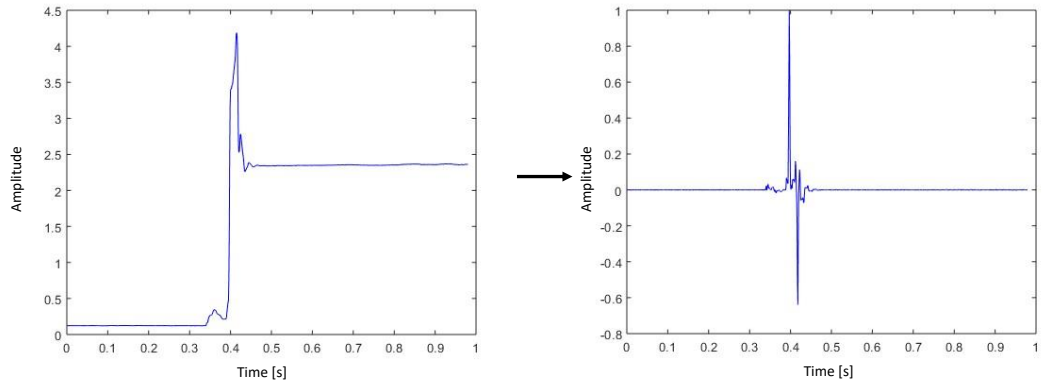

Figure 5.1: Variation with time of the rms current I_{rms} (left) and its derivative I'_{rms}.

Each specific event can be distinguished based on its spectral content and located in a precise time instant. In the above formula, *w* is the window function, and $I'_{rms\,(n)}$ is the sampled signal to be transformed (i.e., the derivative of the rms current value).

The current is processed cyclically at 1 s acquisition intervals, following the described procedure. Each acquisition slot is processed (to calculate rms and the derivative) by adopting an overlap of 500 ms to ensure correct analysis. It is also processed for transient events, which can be fragmented into two successive slots. The STFT is implemented by processing ten-cycle (200 ms) windows with an overlap of 4/5 of the processing window.

To keep track of the type of event (switching ON or OFF), as the spectrograms of a device are often identical for both cases, the spectrogram described in (5.2) is multiplied by the sign of the cumulative sum, evaluated on the rms current signal, as follows:

$$S_N = \sum_{n=1}^{N} \left(I_{rms(n)} - I_{rms(n-1)} \right) \tag{5.3}$$

where $I_{rms\,(n)}$ is the rms value of the current described in (5.1), N is the number of samples, and S_N is the value of the cumulative sum. The final signal $S_{(i,j)}$ can be obtained in the form of a 101 × 26 matrix, as follows:

$$S_{(i,j)} = STFT_{(m,\omega)} \cdot sgn(S_N) = \sum_{n=-\infty}^{\infty} I'_{rms\,(n)} w_{(n-m)} e^{-i\omega n} \cdot sgn\left(\sum_{n=1}^{N} \left(I_{rms\,(n)} - I_{rms\,(n-1)}\right)\right) \quad (5.4)$$

Two examples of the obtained spectrograms are shown in Fig. 5.2 for the case of a microwave oven. The spectrograms obtained by processing the currents flowing through different loads are used as inputs to the neural network described in the following section. This network provides a response every 500 ms, indicating the presence or absence of events in the signal, and the type of device involved.

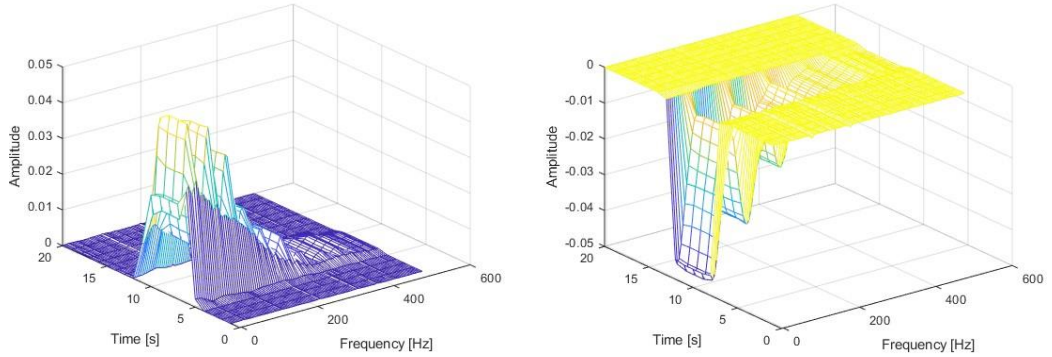

Figure 5.2: Spectrograms obtained during switch ON (left) and OFF (right) of a microwave oven.

5.2 Deep Learning system

Based on the use of complex algorithms, DL provides systems, which are capable of analyzing massive amounts of data, recognizing patterns, and making predictions or decisions without being explicitly programmed to perform these tasks. As introduced in Chapter 2, the DL systems operate by "learning and improving from experience".

This learning ability does not imply that a DL system is capable of understanding what it is analyzing, learning from its experiences, and making decisions based on that understanding. Indeed, the real capabilities of such a system can be summarized considering that if particular behaviors have occurred in the past, it is possible to predict if they can happen again. It also means that if there are no past cases, then there are no predictions. Therefore, the analysis of previous cases is essential for achieving prediction results. Also, the number of cases is generally high.

ANNs are essential parts for the implementation of DL systems. Each node in the ANN receives its input from other nodes or from an external source and calculates its output by applying a function, called activation function (AF), to the weighted sum of its inputs. A bias value is added to this sum. The application of the AF specifically introduces nonlinearities, aiming to emulate the way humans analyze real-world data [179]. The connection between one node and another is performed by a number (weight), which can be positive (one node excites another) or negative (one node inhibits another). The greater the weight, the greater the influence one node exerts on another. If the final sum is above a certain threshold, the node generates an output.

5.2.1 Proposed Convolutional Neural Network

In this work, a particular ANN type, namely, the CNN, is adopted [180] because of its capability of processing complex inputs such as multidimensional arrays. More specifically, CNNs are designed to exploit the intrinsic properties of some 2-D data structures, in which there is a correlation between spatially close elements (local connectivity). The CNNs are capable of reducing the number of operations required by converting the input into modules, which are easier to process.

Thus, compared with ANNs, the number of parameters can be reduced. As a result, CNNs are widely used in the processing of audio and video signals [181]-[183].

To this aim, a CNN suitable to process the current spectrograms was designed. The proposed system, which is shown in Fig. 5.3, includes different layers: an *input*

level (for signal loading), three groups of *convolution*, *Relu*, and *max pooling* layers (for feature extraction from the input), and a group of *flatten*, fully connected, and *softmax* layers, which use data from the convolution layers to generate the output.

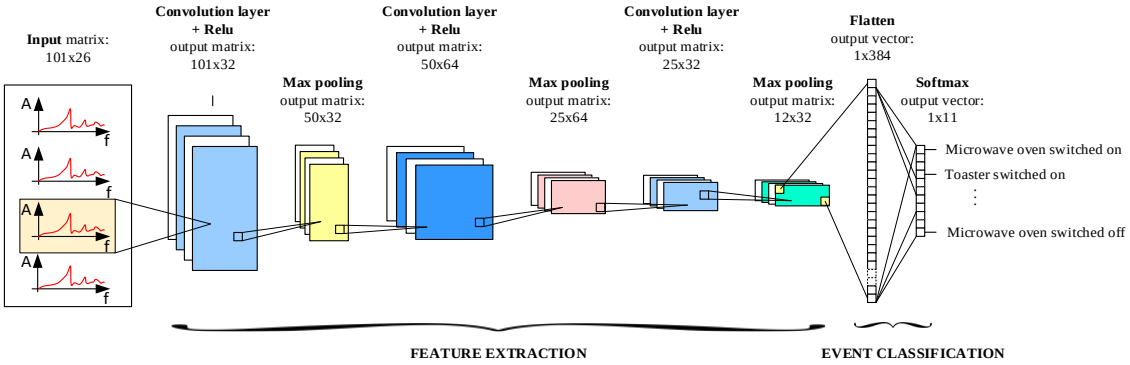

Figure 5.3: Structure of the proposed CNN.

In the proposed CNN:

1) The first layer is the *input* layer, which holds the raw data related to the acquired current. These data are preprocessed through the STFT; each input is a 101 × 26 matrix (frequency × time).
2) The *convolution* layer, which is the core block of a CNN, detects the presence of specific features in the input spectrogram through the application of relevant filters. Instead of processing one matrix element at a time, the convolution layer collects restricted portions (square patches) of data and forwards them through a number of filters. Each of the applied filter seeks a different input parameter, such as a special behavior of the spectrum in a time instant or a particular evolution of a harmonic over time. A filter's spatial dimensions are smaller than the input signal. A filter is also a square matrix, equal in size to the patch, with a set of learnable parameters. Each convolutional layer applies a certain number of filters to its input. In this work, three convolutional layers with different numbers of filters (32 in the first convolution layer, 64 in the second, and 32 in the third) were applied.

   The filters were convolved (slided) across the width and height of the input, and the dot products (between these two matrices) at any position were computed. The result is 2-D arrays (feature maps) that give the responses of the filters at every spatial position.
3) AFs can be either linear or nonlinear. Networks with nonlinear AFs are preferred, since they allow nodes to learn more complex data structures,

even if they require more work in the initial configuration (training). In the proposed network, the AF used at each filter output is the *Relu (z)*, an elementary rectified linear unit (RELU), whose piecewise linear characteristics are presented in Fig. 5.4 [179].

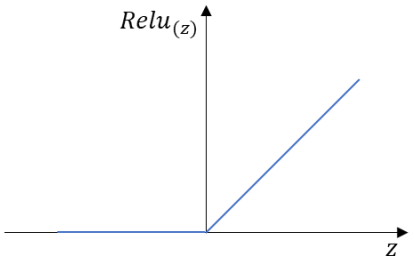

Figure 5.4: RELU AF.

4) The activation maps are fed into a *pooling* (downsampling) layer, which processes one patch at a time, like a convolution. The pooling layer operates on each feature map independently and resizes it spatially using the max operation. The max pooling collects the largest value from one input patch, places it in a new matrix (next to the max values from other patches), and discards the rest of the information contained in the activation maps. This layer is inserted between successive convolution levels to progressively reduce (in this study by approximately 50%) the spatial dimensions of the representation (width, height), keeping the depth intact, to reduce the number of parameters and related calculations.
5) The *Flatten* layer transforms the entire pooled feature map matrix into a single vector (1 × 384).
6) The flattened feature vector is forwarded through a *fully connected* layer, which executes a multiclass classification using the following softmax AF:

$$\sigma_{(Z)\,j} = \frac{e^{z\,j}}{\sum_{k=1}^{K} e^{z_k}} \quad \text{for j} = 1, 2, \ldots K \tag{5.5}$$

This function maps *K* elements of the non-normalized flattened feature vector z_k to a probability distribution over the predicted *K* output classes. The softmax applies an exponential function to each element z_k and then divides it by the sum of all

these exponentials. Each *K* output value represents the probability that an input belongs to that particular class. In this study, five different loads were analyzed.

Therefore, the problem's setting required the definition of ten different classes associated with each device's ON and OFF transients and an additional class (number 5 in Table 5.1), which is related to the "no event occurred" case. The definition of classes is shown in Table 5.1.

Table 5.1: Class definition.

Class	Event
0	Microwave oven switched off
1	Oven switched off
2	Induction hob switched off
3	Toaster switched off
4	Light switched off
5	*No events detected*
6	Light switched on
7	Toaster switched on
8	Induction hob switched on
9	Oven switched on
10	Microwave oven switched on

Each of the 11 values in this output layer corresponds to a class score: the result is the class with the greatest probability. The number 10 indicates the number of events selected and represents the specific application, which was selected as a reference. It is possible to identify a larger number of events by changing the architecture of the CNN.

5.2.2 CNN configuration

As a general remark, it can be considered that the correct functioning of the proposed NILM system can be ensured by optimally designing the architecture of the CNN network (number and type of layers and nodes). It is also necessary to adequately define the AFs and set the network by assigning the appropriate values for the filter weights. After designing the network, as described in Section 5.2.1, the weights of each filter can be defined using the procedure described in the following.

When the CNN operation is started for the first time, the filters are configured by assigning default values to the individual weights. Thus, the results obtained cannot be optimal. The filter weights' adjustment is accomplished through the training process, which consists of two distinct phases [184].

1) *Forward Propagation.* A reference input signal is fed into the input layer. The nodes in the hidden layers apply the defined mathematical operations to these numerical values. The result is sent to the output layer, which generates the final result (classification).
2) *Backward Propagation.* By comparing the generated output with the expected one, the error value can be calculated. This calculation is based on which new (better) filter parameter values are defined.

This cycle is repeated for a new signal, which is obtained from a *reference* set of input data, until the error falls below a predetermined threshold. Obviously, the performance strongly depends on the set of reference examples selected for training. Therefore, these examples must be representative of the real type of the electrical load present. The more data available to the system, the more accurate the load identification will be.

Beyond its unusual name, the so-called training phase is, in fact, an optimization process capable of finding the best solution among all possible ones. During this process, an input data set is mapped to an output data set. The optimization process is based on a function, which represents the error occurring in the network. If the value of this function (which is called cost or loss function) is low (low loss), this means a better system performance. In this study, the function selected was the *logarithmic loss* [109].

At the end of this phase, the network is configured by assigning optimal values for the filter weights based on the analyzed reference data, that is, the values that allow the best event detection and the best load identification. The value of the achieved loss function during configuration was approximately 0.008.

After this phase, tests were conducted to verify the behavior of the system for a set of test signals, other than the reference set. The verification of the correct network configuration was completed by checking whether the system provided the correct answers for inputs other than those considered as a reference. In this work, the reference set adopted consists of 11500 signals and the test set of 2876 signals. The time required for the network training was approximately 7 minutes.

5.3 Experimental results

As part of the development phase, the proposed algorithm was implemented and tested to evaluate its performance with real data.

5.3.1 Proposed system setup

The measurement system includes an Agilent U2542A data-acquisition module with a 16-bit resolution. The sampling frequency was set to 10 kHz. The current signal was acquired using a TA SCT-013 current transducer. The CNN network was implemented on a desktop computer (based on the Windows 10 × 64-bit operating system) using the open-source Python 3.7 from Anaconda [185]. Python is the programming language mostly used in artificial intelligence (AI) applications due to the availability of numerous libraries for continuous data acquisition and processing.

To systematically evaluate the performance of NILM techniques, it is essential to use a set of reference data. The main tests were conducted on signals directly acquired from a real system because of the flexibility regarding both the sampling frequency and the generation of multiple events. Other tests were conducted on signals belonging to a public data set.

The proposed measurement system was installed on a test system, which was designed to generate electrical loads created by domestic users, as part of the "nonintrusive infrastructure for monitoring loads in residential users" research project.

The electrical system, which is located in the Electrical Engineering Laboratory of the University of L'Aquila (I), allows the generation of electrical loads in a single or simultaneous way. These loads correspond to the loads generated by the most common household appliances and are integrated in a structure similar to that of a residential building to reproduce the real problems of conditioning and measurement of the signals.

5.3.2 Results obtained with the acquired signal

Before conducting experimental measurements, the current measurement channel was calibrated using the Fluke 6100A power-supply standard. More specifically, a reference current was generated and applied to the SCT-013 current transducer. The current was acquired through the Agilent U2542A data-acquisition system, and the data were processed to calculate the rms value. In this way, the entire signal acquisition and processing path were tested. The system was calibrated

with ten different current values, ranging from 2 A to 20 A. The maximum uncertainty obtained was approximately ±1.3%.

Subsequently, the performance of the NILM system was assessed by conducting acquisitions, during which various loads were turned ON and OFF for a total of over 519 events.

Regarding the NILM systems, no standard and consolidated techniques can be found in the literature to evaluate the performance of event detectors. Since the purpose of an NILM system is to disaggregate consumption for each of the devices in question, their performances were analyzed to verify the achievement of these objectives, which in summary are correct identification and classification of the events.

The first thing verified was the ability to correctly identify the ON and OFF events, which were also performed in rapid succession, and to correctly classify the device that produced a particular event. The proposed system was found to be capable of correctly identifying the insertion of loads, even by performing maneuvers at very short time intervals, up to approximately 500 ms. An example of the acquired signal representing the current variation for a 1 min window is presented in Fig. 5.5. This signal was extracted from the overall acquisition process. The effective value of the measured current and the relative system responses are also shown in this figure. It can be observed that the system is capable of detecting all the events.

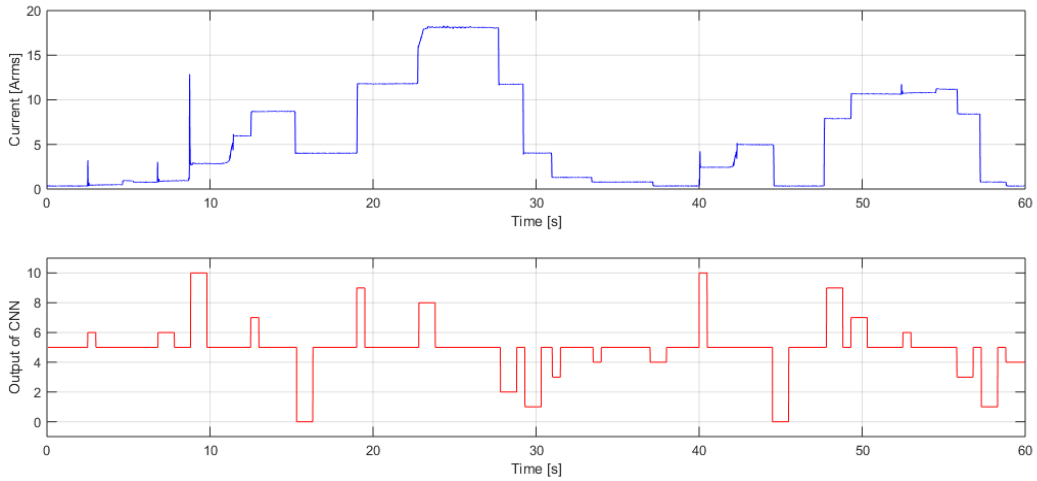

Figure 5.5: Sequence of events: variation in the rms current (above) and detected events (below).

Since the electrical load signature proposed in this work is based on the transitory characteristics, the system defined is not capable of classifying two different devices when the events associated with them are exactly superimposed.

Tests were conducted to verify the system's ability to detect temporally close events. Fig. 5.6 shows that a toaster's switch on occurs approximately 400 ms after a microwave oven's switch on and the consequent system response. It is possible to observe how the system perfectly distinguishes the two events and classifies them correctly (events 10 and 7). A third event in the interval between 6 s and 7 s can be observed in Fig. 5.6.

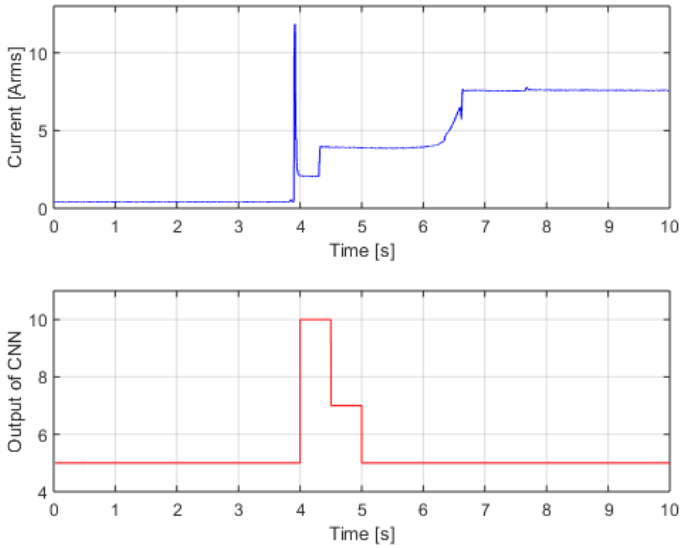

Figure 5.6: Switch ON of a toaster 400 ms after the switch on of a microwave oven (above) and detected events (below).

This transient is normally produced by the microwave oven under examination approximately 3 s after its start. The double transient is very common in many household appliances. In an oven, for example, the electronic section is activated first followed by the heating section. The proposed system was configured to filter this transient, classifying it as a nonevent (5), being linked to the first correctly recorded insertion. It was verified that the system is capable of identifying nearby events up to 300 ms.

The percentage of events, which were correctly detected, was compared with the total number of real events by adopting the following definitions.

1) Number of true power-ON events: ON_T
2) Number of true power-OFF events: OFF_T
3) Number of windows with no events: NE_T

4) Number of power-ON events identified: ON_i
5) Number of power-OFF events identified: OFF_i
6) Number of no events identified as no events: NE_i
7) Number of correctly classified power-ON events: ON_c
8) Number of correctly classified power-OFF events: OFF_c
9) Error in the identification of power-ON events

$$E_{ident,on}\% = \frac{ON_i - ON_T}{ON_T}100 \tag{5.6}$$

10) Error in the identification of power-OFF events

$$E_{ident,off}\% = \frac{OFF_i - OFF_T}{OFF_T}100 \tag{5.7}$$

11) Error in the identification of no events

$$E_{ident,ne}\% = \frac{NE_i - NE_T}{NE_T}100 \tag{5.8}$$

12) Error in the classification of power-ON events

$$E_{class,on}\% = \frac{ON_c - ON_T}{ON_T}100 \tag{5.9}$$

13) Error in the classification of power-OFF events

$$E_{class,off}\% = \frac{OFF_c - OFF_T}{OFF_T}100 \tag{5.10}$$

The absolute values relating to these parameters are presented in Table 5.2.

From Table 5.2, it can be observed that the system is capable of perfectly identifying all the OFF events. An error of approximately 3% was recorded regarding the ON events. This error value is essentially linked to the multiple oscillations of the current signal produced by some devices during the activation phase. Regarding the classifier performance, errors of 1.84% were recorded during the ON phases and 1.21% during the OFF phases, respectively.

For a better evaluation of the system performance, precision, recall, and F1-score metrics were also used [186]. These parameters, already mentioned in Chapter 3 in Section 3.4.1, were obtained using the number of true positive (TP), false positive (FP), true negative (TN), and false negative (FN) as follows:

$$Precision = \frac{TP}{TP + FP} \tag{5.11}$$

$$Recall = \frac{TP}{TP + FN} \tag{5.12}$$

$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{5.13}$$

$$FPR = \frac{FP}{FP + TN} \tag{5.14}$$

$$FPP = \frac{FP}{TP + FN} \tag{5.15}$$

The calculated results are presented in Table 5.3.

Table 5.2: Errors measured during the processing of the acquired signals.

Number of processed windows	2883
Number of total events	519
Number of true power-on events ON_T	272
Number of true power-off events OFF_T	247
Number of windows with no events NE_T	2364
Number of power-on events identified ON_i	281
Number of power-off events identified OFF_i	247
Number of no-events identified as no-events NE_i	2355
Error in the identification of power-on events $E_{ident,on}(\%)$	3.31%
Error in the identification of power-off events $E_{ident,off}(\%)$	0.00%
Error in the identification of no events $E_{ident,ne}(\%)$	0.38%
Number of correctly classified power-on events ON_c	267
Number of correctly classified power-off events OFF_c	244
Error in the classification of power-on events $E_{class,on}(\%)$	1.84%
Error in the classification of power-off events $E_{class,off}(\%)$	1.21%

Table 5.3: Scores achieved with the acquired signals.

TP	*TN*	*FP*	*FN*	*Precision*	*recall*	*FPR*	*FPP*	*F1-score*
518	2354	10	1	0.981	0.998	0.004	0.019	0.989

The system was capable of correctly classifying 511 of 519 overall events. The obtained basic accuracy during classification [186], which is defined as

$$Acc\% = \frac{Correct\ matches}{Total\ \ possible\ matches} 100 \tag{5.16}$$

is 98% with the acquired signals.

The errors obtained during the classification phase can be tabulated using the error matrix [187]. This is shown in Table 5.4. Each column of the matrix represents instances in a predicted class, whereas each row represents instances in an actual class.

Table 5.4: Error Matrix

Actual events \ Events found	Microwave oven switched off	Oven switched off	Induction hob switched off	Toaster switched off	Light switched off	*No events detected*	Light switched on	Toaster switched on	Induction hob switched on	Oven switched on	Microwave oven switched on	Microwave oven switched on
Microwave oven switched off	47	0	0	0	0	0	0	0	0	0	0	
Oven switched off	0	52	2	0	0	0	0	0	0	0	0	
Induction hob switched off	0	1	31	0	0	0	0	0	0	0	0	
Toaster switched off	0	0	0	54	0	0	0	0	0	0	0	
Light switched off	0	0	0	0	60	0	0	0	0	0	0	
No events detected	0	0	0	0	0	2354	10	0	0	0	0	
Light switched on	0	0	0	0	0	1	75	0	2	0	0	
Toaster switched on	0	0	0	0	0	0	0	53	0	2	0	
Induction hob switched on	0	0	0	0	0	0	0	0	35	0	0	
Oven switched on	0	0	0	0	0	0	0	0	0	57	0	
Microwave oven switched on	0	0	0	0	0	0	0	0	0	0	47	

The second type of tests is related to the measured duration between the ON and OFF events, being linked to the energy consumed. The results shown in Fig. 5.7 show a rapid succession of switch ON (event 9) and OFF (event 1) for the oven. The obtained results, which are tabulated in Table 5.5, show a maximum relative error of 2.68%. However, it should be noted that this value is related to the reduced activation time (11.2 s), which was produced during the test. With longer intervals, for example 49.8 s, the relative error is reduced to 0.40%.

The absolute error essentially depends on the duration of the window used for processing the STFT. The reduction of this window results in a reduction in this

error. However, the processing time increases. The value of the window duration selected allows a good compromise, considering that the activation times of the loads are normally much greater than those selected in the test.

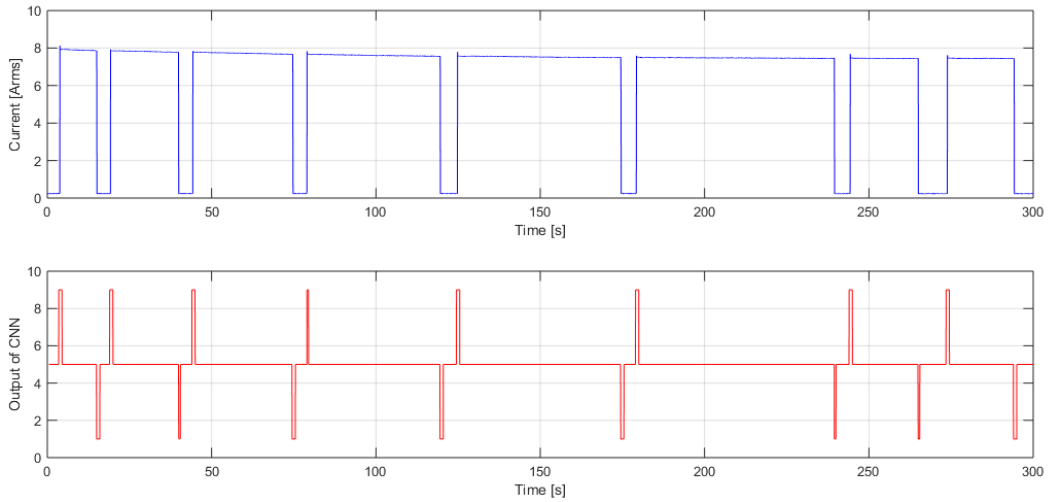

Figure 5.7: Oven switch ON and OFF.

Table 5.5: Errors measured during a succession of ON and OFF switching.

Event	Actual time	Measured time	Actual time interval [s]	Measured time interval [s]	Error [%]
1	0:03.8	0:04.0	11.2	11.5	2.68
2	0:15.0	0:15.5			
3	0:19.2	0:19.5	20.8	21.0	0.96
4	0:40.0	0:40.5			
5	0:44.2	0:44.5	30.5	30.5	0.00
6	1:14.7	1:15.0			
7	1:19.0	1:19.5	40.6	40.5	−0.25
8	1:59.6	2:00.0			
9	2:04.8	2:05.0	49.8	50.0	0.40
10	2:54.6	2:55.0			
11	2:59.2	3:00.0	60.3	60.0	−0.50
12	3:59.5	4:00.0			
13	4:04.3	4:04.5	20.7	21.0	1.45
14	4:25.0	4:25.5			
15	4:33.8	4:34.0	20.4	20.5	0.49
16	4:54.2	4:54.5			

The system's performance was further evaluated by investigating its behavior with respect to distinguishing currents flowing in small loads in the presence of significant current values. To this purpose, Fig. 5.8 shows an acquisition in which a lamp is switched on (0.54 A) while more than 10 Arms are already flowing in the

system (this represents 5% of the total load). The system proved to be efficient even under these operating conditions by identifying the event and correctly classifying the device.

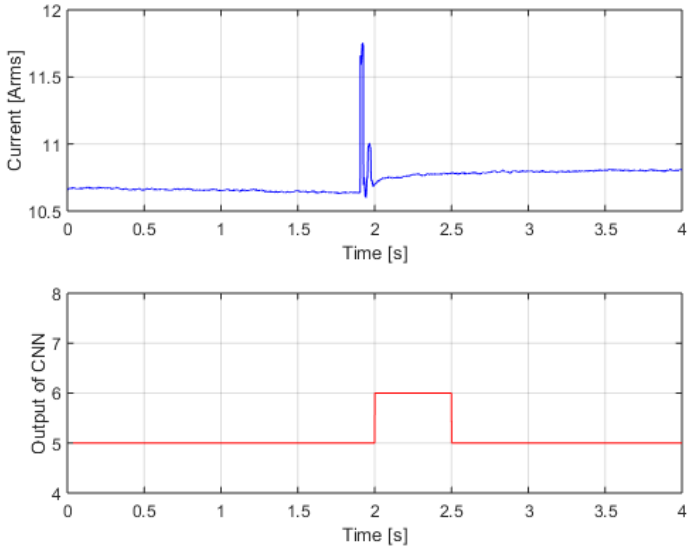

Figure 5.8: Lamp switch ON.

5.3.3 Results obtained with the BLUED dataset

To make comparisons between the obtained results, tests were also conducted using the Building-Level fUlly-labeled data set for Electricity Disaggregation (BLUED), which is a residential electricity-usage public data set. This data set includes voltage and current measurements for a single-family house in the United States, sampled at 12 kHz for an entire week [188].

The current signal, which was extracted from the data set, was processed as described previously. Specifically, the spectrograms of 59587 windows of 1-s duration were processed by the CNN, and 59587 output values were obtained. This data set was randomly divided into two parts: 80% for network training and 20% (11918 windows) for its performance evaluation. In the windows analyzed, 3887 switch-ON events and 4014 switch-OFF events were observed, whereas 4017 windows were free of events.

For the elaboration of the BLUED data set, the CNN was modified by extending the classification of Table 5.1 to 69 different classes, to detect the ON and OFF states of 34 different types of devices. The obtained results are presented in Table 5.6. The scores achieved with the BLUED data set are tabulated in Table 5.7.

The obtained basic accuracy (5.16) with the BLUED data set is 87.9%. The data processing value of the data set is lower than that obtained by processing the signals acquired. This is due to the greater number of devices the system must identify.

Table 5.6: Errors measured during the processing of the BLUED dataset.

Number of processed windows	11918
Number of total events	7901
Number of true power-on events ON_T	3887
Number of true power-off events OFF_T	4014
Number of windows with no events NE_T	4017
Number of power-on events identified ON_i	3890
Number of power-off events identified OFF_i	4006
Number of no-events identified as no-events NE_i	4022
Error in the identification of power-on events $E_{ident,on}$ *(%)*	0.08%
Error in the identification of power-off events $E_{ident,off}$ *(%)*	0.20%
Error in the identification of no events $E_{ident,ne}$ *(%)*	0.12%
Number of correctly classified power-on events ON_c	3569
Number of correctly classified power-off events OFF_c	3377
Error in the classification of power-on events $E_{class,on}$ *(%)*	8.18%
Error in the classification of power-off events $E_{class,off}$ *(%)*	15.87%

Table 5.7: Scores achieved with the BLUED dataset.

TP	*TN*	*FP*	*FN*	*precision*	*Recall*	*FPR*	*FPP*	*F1-score*
7890	4011	6	11	0.998	0.998	0.001	0.001	0.998

Chapter 6

A low sampling rate non-event-based NILM system

The third proposed system is a non-event-based system that processes the sampled "low-frequency" absorbed active power signal. The proposed solution is based on a CNN, implemented on an Arm Cortex-M7 microcontroller. Specifically, it is a sequence-to-point approach, first proposed in [113], which demonstrated a significant performance improvement over previously used approaches. The algorithm has been tested on several public datasets; in particular, its generalization capability was also tested [122]. Several public datasets were used in [122]. This approach was tested by providing the system with data on houses belonging to datasets different from the one presented in the training phase. However, these datasets do not allow the full potential of these systems to be evaluated. This is due both to the randomness of the sub-monitored loads in the different homes within the same dataset, and to the reduced flexibility of sampling frequencies, that are always fixed.

Another major problem, from a metrological point of view, is related to the great difficulty, if not impossibility, of knowing the uncertainty associated with the measurements of the quantities (current, power) of the datasets.

Cloud-based energy disaggregation systems generally consist of sending energy consumption data to a remote server, where it is processed and made available for user access from anywhere via a Web browser. These solutions are convenient because it does not require dedicated hardware to process the data, even if some disadvantages are present. For example, sending data to a remote server involves some latency and depends on the quality of the Internet connection, thus risking

being subject to service interruptions. In addition, processing data remotely involves greater complexity in data management and security.

In contrast, an Edge solution, such as the one proposed, consists of processing energy consumption data directly on the device installed at the facility to be monitored. Then the processed data can be accessed locally or transmitted to a remote server via an Internet connection. This approach has several advantages over Cloud solutions. For example, Edge solutions are able to process data in real time, eliminating the latency present in Cloud solutions. They are easier to install and manage, as they do not require a reliable Internet connection. This also makes these types of systems cheaper than their Cloud counterparts even if the more knowledge in managing real-time systems is required. In addition, data access is limited only to authorized devices at the monitoring site, meaning there are no costs associated with transmitting data or maintaining a remote server.

Ultimately, Edge solutions for NILM offer greater reliability, privacy, security, and convenience than Cloud solutions.

This chapter therefore proposes an Edge system for energy disaggregation that can be installed in an electrical system, without any a priori knowledge of the electrical system and its loads.

Moreover, a comprehensive architecture for a measurement system designed for the monitoring and acquisition of various electrical quantities generated by household appliances is presented. The system captures the aggregate consumption of the electrical system and individual appliances, and wirelessly transmits this information to a central concentrator. The captured data also includes the disaggregated information produced by the NILM system, which effectively provides both individual and overall consumption. Utilizing this data, the performance of NILM systems can be thoroughly characterized and evaluated. The data are metrologically suitable, being acquired and processed with a single-phase board meter certified in class 0.2.

The implementation of both the measurement architecture and data processing software, as well as the results obtained from the system are described. The system was installed in two different houses in Italy for a period of six months each. Therefore, the testing phase lasted a total of twelve months, from January 2022 to February 2023 (inclusive).

6.1 NILM as a nonlinear regression problem

As previously introduced, the model implemented in the system proposed in this chapter is a non-event-based DL model. This model is also based on a supervised learning mode. This category of systems involves showing both input samples and expected output samples during the training phase. This is therefore referred to as labeled datasets. For non-event-based NILM systems, a dataset is labeled when both the time sequence of the aggregate absorbed power signal and the time sequences related to the power absorbed by individual appliances (or related to their ON/OFF state) are provided. The data pairs (X_t, Y_t) are therefore available, where X_t indicates the reading of the aggregate absorbed power, and Y_t indicates the reading of the absorbed power at the appliance-level. The goal of a supervised non-event-based model is to learn the relationship between X and Y. In this way, the problem formulated in (6.1) can be approached as a nonlinear regression.

$$X = f(Y) \tag{6.1}$$

The approach used in this work is of the sequence-point type. Given an aggregate power reading window, an ANN is trained for it to exclusively predict the midpoint of an appliance-level power reading window. In this way, the overall time series is obtained through sliding window processing.

By indicating with t the generic instant of time and with W the length of the window, for each window $X_{t:t+W-1}$ is estimated the power absorbed by the appliance monitored in the central point of the window $Y_{t+\frac{W}{2}}$.

This approach is based on the assumption that the central point $Y_{t+\frac{W}{2}}$ can be represented as a nonlinear regression of the input window $X_{t:t+W-1}$, and thus that the estimate of the power absorbed by a household appliance in a certain instant of time should not only be influenced by past power readings, but also by future ones.

Many of the models used to represent the relationship f between X series and Y series are subject to important limitations. Models such as FHMM are strongly influenced by the presence of unknown appliances, base load, and noise. Implementation of these models therefore requires explicit modeling of these variables.

In contrast, the use of DL models does not need to model them explicitly. In fact, DL models separate the consumption profile of an appliance by treating everything else as background.

Desired characteristics such as individual appliance consumptions, transitions between ON and OFF states, and duration of operation are learned automatically by the neural network and do not require any manual extraction.

In fact, this process does not require the use of specific information about consumption sources or their power profiles, which means that the model is able to generalize to new consumption sources that might appear over time. Thus, this model can be used to separate independent signals associated with different consumption sources in a wide range of situations without the need for specific information about consumption sources or their power profiles.

6.1.1 Model configuration

The implemented sequence-to-point model consists of convolutional layers, so it falls into the category of CNNs. As already explained, CNNs are particular ANNs designed to exploit the inherent properties of certain two-dimensional data structures in which there is a correlation between spatially close elements (local connectivity).

The same process can be applied to one-dimensional data sequences. A one-dimensional (1D) CNN is very effective for deriving features from a fixed-length segment of the overall dataset, where it is not important where the feature is located in the segment. CNNs work the same whether they are 1-, 2-, or 3-dimensional. The difference lies in the structure of the input data and how the filter and the convolution kernel acts on the data.

In this work, a suitable 1D CNN was implemented to process a time sequence of the aggregate power signal, in order to predict the midpoint of the time sequence of the absorbed power at the appliance level. Since the only dimension in a time series is time, the kernel will flow in only one direction, as shown in Fig. 6.1.

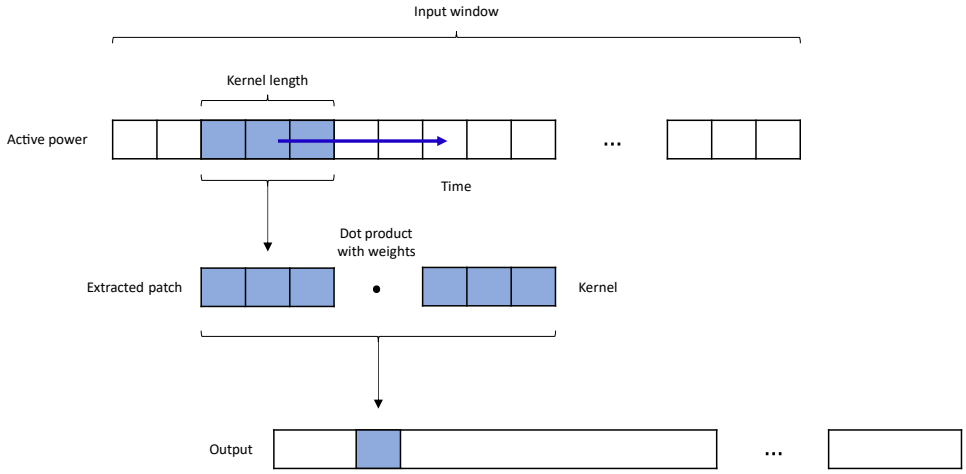

Figure 6.1: Operation of the first 1D Conv layer on a time sequence.

A 1D convolutional layer extracts, for each kernel, 1D local patches (i.e., subsequences) of the original sequence through a sliding window processing. It then applies an identical transformation to each patch. For each kernel, since the same transformation is applied to each patch, a pattern learned at a certain position in the sequence can also be recognized at a different position. This makes the 1D CNN translation invariant (for temporal translations).

The proposed 1D CNN involves the following building block:

1) The first is the *input layer*. In this layer, data are pre-processed through a sliding window technique such that each input contains 599 samples of the aggregate active power reading. The data are sampled at a sampling rate of 1/8 Hz, so each input sequence covers a time interval of 4792 seconds.

2) A *first 1D convolutional layer*. In the first 1D convolutional layer, 30 filters (or kernels) of length 10 (kernel size) are defined, which allow ANN to learn 30 different features. In these layers, the step (stride) with which the kernel moves along the input sequence is also defined. The number of filters that are applied in each layer is a hyperparameter, which is chosen by the programmer. The features that each filter will learn, and thus the weights to assign to the filter, are the result of training. In this work, a stride equal to 1 has been defined. A kernel of size 10 moving along an input sequence of size 599 with a stride equal to 1 will produce an output sequence of 590 elements. However, a padding process was used in order to fill the input sequence with a certain number of zeros at

the beginning and end of the sequence, in order to output the same number of elements as the input sequence (599). The output of the first 1D convolutional layer is thus an array of 30 x 599 neurons. That is, 30 output sequences resulting from applying 30 filters on the input sequence.

3) The *second 1D convolutional layer*. The result of the first 1D convolutional layer is directly fed into the second layer. Also in this layer, 30 filters are defined to be trained. Although the input of this layer is now two-dimensional instead of one-dimensional (a 30 x 599 matrix), the transformation applied by this layer is still a 1D convolution. Therefore, the kernels will move along a single (temporal) direction. The dimension chosen for the kernels in this layer is 8. The kernels in this case will no longer be vectors of length equal to the imposed kernel size (1 x 8) but will be a 30 x 8 matrix. Fig. 6.2 illustrates the process of kernels in the second 1D convolutional layer.

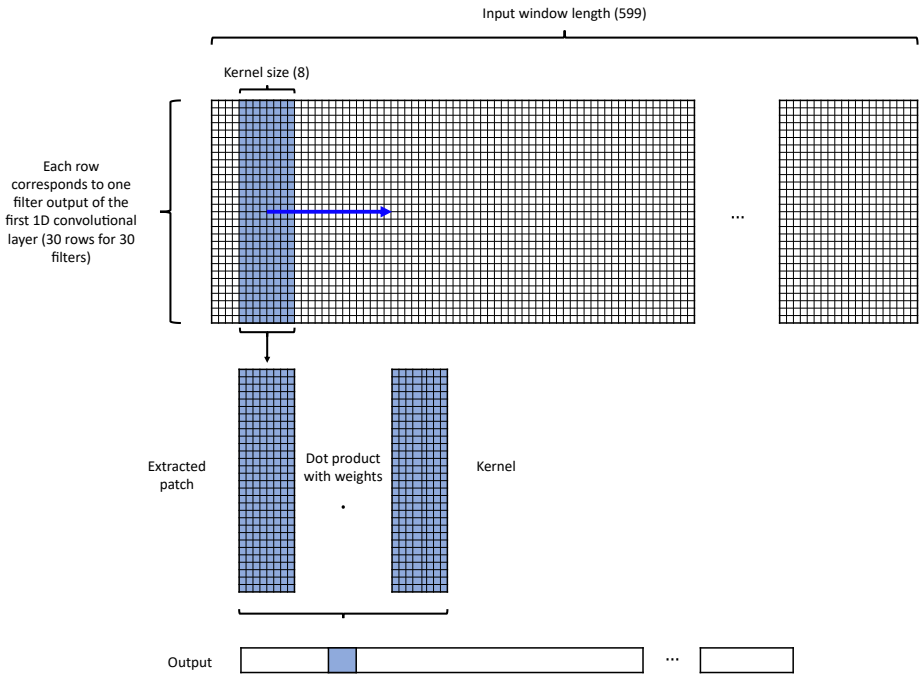

Figure 6.2: Operation of a filter of the second 1D Conv layer.

4) *Third, fourth and fifth 1D convolutional layers*. Three more 1D convolutional layers were added to learn higher-level features. The kernel number of layers three, four and five is 40, 50 and 50 respectively. The kernel size is 6, 5 and 5, respectively. The stride was also kept equal to 1

for these additional three layers. The padding process was provided for all convolutional layers. All neurons in the five 1D convolutional layers predict using the ReLU activation function.

5) A *flatten layer*. This is a layer that transforms the entire output matrix of the fifth and final 1D convolutional layer into a single vector (1 × 29950).
6) A *dense layer*. This layer reduces the output dimension from 29950 to 1024 from the flatten layer. The neurons in this layer also use the ReLU activation function.
7) An *output layer*. This layer provides the value of the midpoint of the appliance power reading, therefore this layer has a single neuron. This neuron receives the weighted sum of the 1024 output elements from the previous layer as input. It then applies a linear activation function, which is effectively the same as applying no activation function at all.

Fig. 6.3 shows the overall structure of the implemented 1D CNN.

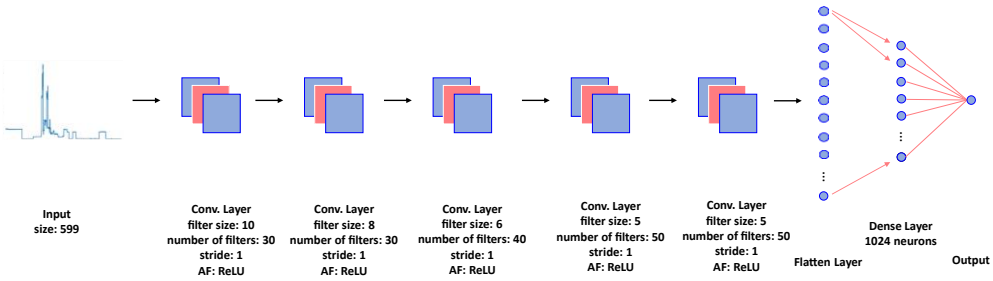

Figure 6.3: CNN configuration.

6.1.2 Training settings

In the last decade, alongside the proposals for new NILM algorithms, a fair amount of public datasets have been made available to allow researchers to compare the performance of their systems. These datasets differ in sampling frequency, quantity of homes monitored, availability or not of sub-monitored data (i.e. measurements made directly on household appliances or other loads). Even when sub-monitored data is available, it is not always synchronized with the aggregate power measurement.

The requirements that a good dataset should have, in order for it to evaluate the performance of a DL-based NILM systems, are:

1) sampling frequency of 1 Hz or more, which allows to evaluate the impact of the latter on performance, being able to obtain all the lowest frequencies starting from the original data;
2) synchronous measurements of aggregate and appliance-level quantities for all quantities;
3) sufficient number of houses (at least > 1);
4) sufficient class number of household appliance;
5) acquisition period sufficiently long, in order to have a sufficient amount of data for training.

To date, the most complete datasets in this regard are ECO [189] and ENERTALK [190]. ECO provides aggregate measurements of active power, voltage, current and power factor at 1 Hz for 6 different homes and 18 different classes of appliances; however only active power at appliance-level is available. ENERTALK, on the other hand, provides both aggregated and appliance-level active and reactive power measurements at 15 Hz for 22 homes, however it has a small number of measured appliance classes.

The main datasets used for training DL-based NILM systems and their characteristics are listed below:

- The Reference Energy Disaggregation Data Set (REDD) [88], which provides both high frequency data, i.e. voltage and current measured at 15 kHz, and low frequency, measured at a frequency of 1 Hz.
- The United Kingdom domestic appliance-level electricity (UK-DALE) dataset [191], which contains data from 5 houses providing both aggregate power and appliance level measurements at 1/6 Hz.
- The REFIT dataset [149] provides aggregate and appliance-level power measurements for 21 homes at a sampling rate of 1/8 Hz. However, not all 21 homes have the most common appliances on which NILM systems are trained.

The REFIT dataset is the largest among the datasets mentioned. Therefore, although the low sampling rate does not allow for making assessments about its influence on performance, the wide availability of examples allows for a sufficiently robust NILM system. The training phase was consequently carried out using the latter dataset. The proposed system utilizes a CNN to identify and recognize individual household appliances. The focus was on developing a NILM system capable of separating the electrical power consumption of the three major household loads - dishwasher, washing machine, and fridge - from the total consumption. This decision was made as these loads are among the most commonly targeted by NILM system developers for disaggregation services in the market [17].

It is therefore necessary to train the DL model individually, training it only with data from that household appliance. The REFIT dataset contains measured consumption data for 21 UK homes, but since not all of them contain data for the three loads mentioned above, Table 6.1 shows the homes used to train each model and the total number of samples available.

Table 6.1: Houses from the REFIT dataset used for the training phase.

	Training set		**Validation set**	
	Houses	**Samples**	**Houses**	**Samples**
Dishwasher	5, 7, 9, 13, 16, 20	35984834	18	5007721
Washing machine	2, 5, 7, 8, 9, 15, 16, 17	49588126	18	5007721
Fridge	2, 5, 9, 15	25559502	12	5859544

Table 6.1 also shows the houses and the relative number of samples used as the validation set. The validation set plays a crucial role in the training process of DL models, as it allows to evaluate the model's ability to generalize to unseen data. The validation set is especially vital as it provides a continuous assessment of the model's performance during training, by monitoring it at the end of each epoch. This helps to avoid a common problem in DL called overfitting [192], where the model becomes too tightly fitted to the training data, hindering its ability to perform well on new data.

Prior to utilizing the data for training, it is crucial to preprocess it. The preprocessing involves normalizing the data using the (6.2).

$$\frac{x_k - \bar{x}}{\sigma_x} \tag{6.2}$$

where x_k is the k-th sample, $\bar{x}$ is the mean of the aggregate or appliance-level power reading and σ_x is the standard deviation of the aggregate or appliance-level power reading. Once the data has been normalized, it can be fed into the models for training.

The CNN networks were implemented on a desktop computer (based on the Windows 10 × 64-bit operating system) using Tensorflow [193] for the model development and training. The adopted cost function, implemented in Tensorflow, is the mean squared error (MSE) (6.3), applied to each batch during the training phase.

$$MSE = \frac{1}{N}\sum_{i=1}^{N}\left(Y_i - \hat{Y}_i\right)^2 \tag{6.3}$$

where Y_i is the actual value, $\hat{Y}_i$ is the predicted value and N is the number of processed samples.

The parameters of the CNN are update after one iteration of every batch of data. The batch size chosen for training is 1000, so the neural network parameters were updated every 1000 samples. Each model has been trained for 10 epochs.

The Adam optimizer [194] was selected to drive the models training process. Adam is a highly efficient optimization algorithm that has proven to be a popular choice among practitioners in the field of DL. It operates by utilizing an adaptive learning rate that is dynamically adjusted based on the mean and variance of the gradients of the cost function, with respect to the weights of the ANN. During the training phase, Adam continuously monitors the mean and variance of the gradients, allowing it to effectively fine-tune the learning rate. By increasing the learning rate for slowly changing weights and decreasing it for rapidly changing weights, Adam effectively eliminates oscillations and accelerates convergence towards the global minimum solution.

6.2 Architecture of the proposed system

The proposed architecture is based on a distributed data acquisition system, communicating on a Wi-Fi network.

The first device developed (NILM System in Fig. 6.4) is able to measure the aggregate active power and disaggregate it through the model described in Section 6.1. In addition to measuring active power, this system also measures reactive power, voltage, and current, which, however, do not come into play in the performance evaluation of this NILM system. This NILM system consists of:

- a measurement unit, consisting of an EVALSTPM32 board;
- a processing unit, consisting of a NUCLEO-H743ZI2 board;
- an ESP32 Wi-Fi module for connection to a WLAN.

The main device has been placed immediately downstream of the general power meter located at the user's connection point, so that the aggregate active power can be measured.

In addition, a number of ad hoc power meters have also been developed, consisting of:

- a measuring unit, consisting again of a EVALSTPM32 board;
- a USR-W610 Wi-Fi module for connection to a WLAN.

These appliance-level power meters allow measurement of electrical quantities (active power, reactive power, rms voltage and rms current) related to the operation of individual household appliances.

A wireless local area network (WLAN) has been established using a star topology, where both the NILM system and the appliance-level power meters are connected to the concentrator via Wi-Fi network. More specifically, the implemented communication and data archiving infrastructure, concisely schematized in the right part of the Fig. 6.4, consists of:

- an access point (AP), which creates the wireless network where all nodes are connected;
- an INTEL NUC NUC5i7RYH system, which is the master of the network where a Python script manages the nodes connected to the network, downloading the measurement data and storing the data on a MySQL database;
- an external hard disk for storing the MySQL database.

Furthermore, a web server has been developed based on Node-RED to check the whole system and plot few point of measurement data.

The proposed system is shown in Fig. 6.4.

To facilitate clear and concise understanding of the connections involved that are schematized in Fig. 6.4, the Fig. 6.5 provides a detailed view of the EVALSTPM32 board's phase and neutral conductor connections for voltage and current measurements. In the voltage measurement circuit, the N terminal was deliberately left unconnected to the neutral conductor. This was because the shunt for the current measurement is already placed at the same potential as the N terminal, which eliminates the need for a separate connection.

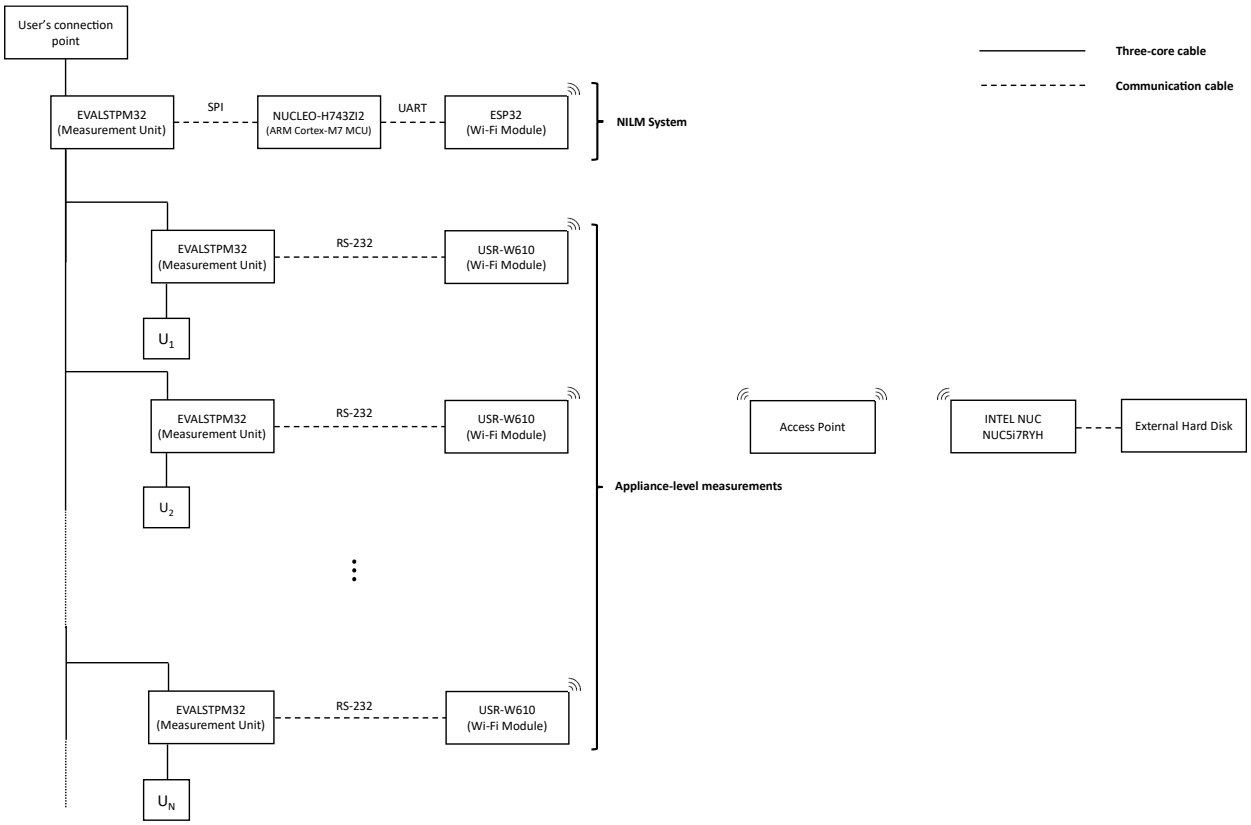

Figure 6.4: The proposed architecture.

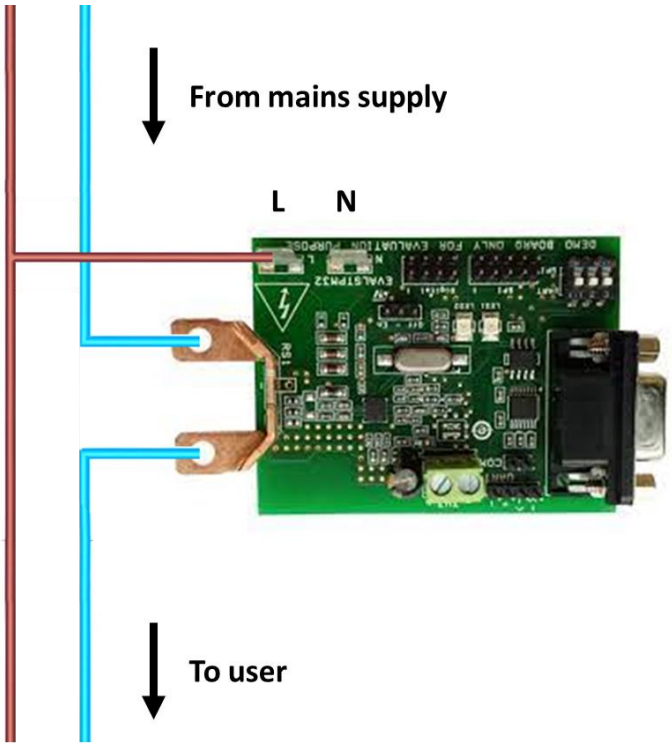

Figure 6.5: Pin connections of the EVALSTPM32 board for voltage and current measurement.

6.2.1 NILM system and appliance-level power meters

The pre-trained DL models, described in Section 6.1, were uploaded to an NUCLEO-H743ZI2 board. This is a high-performance microcontroller from STMicroelectronics, based on the ARM Cortex-M7 architecture. This board is part of the STM32H7 series and offers a range of advanced features, including a floating-point unit, hardware encryption, and up to 2MB of flash memory. The NUCLEO-H743ZI2 also includes a variety of communication interfaces, including Ethernet, USB, CAN, and various serial ports. It is designed for use in a wide range of applications, including industrial automation, motor control, and consumer electronics. The microcontroller can be programmed using a variety of integrated development environments (IDEs) and supports a range of development tools and software libraries provided by STMicroelectronics.

The IDE used for this work is STM32CubeIDE. The models were implemented through X-CUBE-AI, that is an expansion package dedicated to AI projects running on STM32 Arm® Cortex®-M-based MCUs. The X-Cube AI core engine, schematically shown in Fig. 6.6, offers an NN mapping tool to create and implement a pre-trained DL model for embedded systems with limited hardware resources.

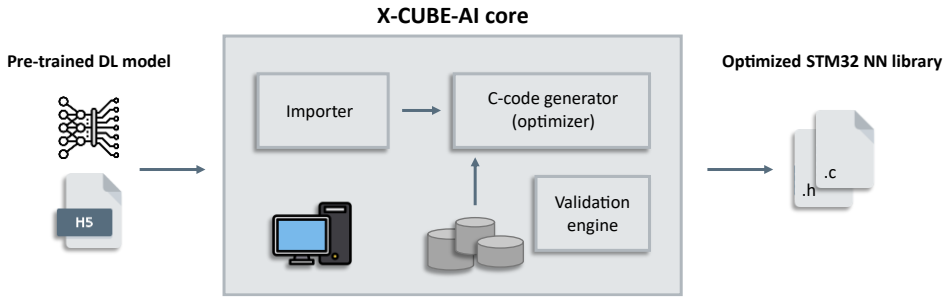

Figure 6.6: The X-Cube AI core engine.

The generated STM32 NN library can be integrated into an IDE project or makefile-based build system. The code generator quantizes weights, bias, and activations from floating point to 8-bit precision and maps them onto a specialized C implementation for supported kernels. This technique aims to reduce the model size, improve CPU and hardware accelerator latency, and reduce power consumption without sacrificing model accuracy.

A validation mechanism is provided to compare the accuracy of the generated model with the uploaded DL model using the same input tensors (fixed random inputs or custom dataset). The scheme of the validation engine is shown in Fig. 6.7.

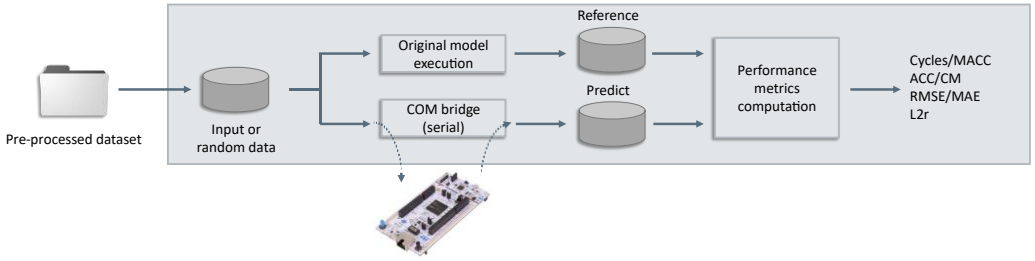

Figure 6.7: AI validation firmware embedding the tested C model.

The NUCLEO-H743ZI2 board receives input data for models via an EVALSTPM32 board.

The measurement unit is realized by an EVALSTPM32 board. The EVALSTPM32 board is a class 0.2, single-phase meter, that acquires the power line current with a shunt transducer. The board has SPI/UART pins available to interface a microcontroller for application development. The EVALSTPM32 provides instantaneous voltage and current waveforms and calculates RMS values of voltage and currents, as well as active, reactive and apparent power and energies. The EVALSTPM32 is a mixed signal IC family consisting of an analog and a digital section. The analog section consists of up to two programmable gain low-noise low-offset amplifiers and up to four 2nd order 24-bit sigma-delta ADCs, two bandgap voltage references with independent temperature compensation, a low drop voltage regulator and DC buffers. The digital section consists of digital filtering stage, a hardwired DSP, a digital front-end (DFE) and a serial communication interface. Power data registers inside the board can supply instantaneous measurements or filtered measurements. The bandwidth of wide band waveforms is 3.6 kHz, which means up to the 72nd harmonic of a 50 Hz signal.

To connect the EVALSTPM32 to the NUCLEO-H743ZI2, the SPI interface is used, configured with a clock frequency of 10 MHz and full duplex transmission mode. The configuration of the SPI on the NUCLEO-H743ZI2 is done using the STM32CubeIDE HAL library. Once the SPI is configured, the firmware on the NUCLEO-H743ZI2 requests the reading of active power, reactive power, rms current, and rms voltage data from the EVALSTPM32 using the SPI protocol.

Communication takes place when the NUCLEO-H743ZI2 sends the request and the EVALSTPM32 replies with the requested data. Once the data is read, the NUCLEO-H743ZI2 processes the active power data using the models described above, to obtain appliance-level active power information. Once the data is processed, the NUCLEO-H743ZI2 uses the ESP32 module to connect to the WiFi network and send the data to the concentrator, using the response-response protocol based on TCP. The ESP32 module is configured as a client and communication

occurs asynchronously. A response-response protocol allows for bidirectional communication between the NUCLEO-H743ZI2 and the concentrator, with the ability to send data in both directions. The described NILM system is shown in the photo of Fig. 6.8.

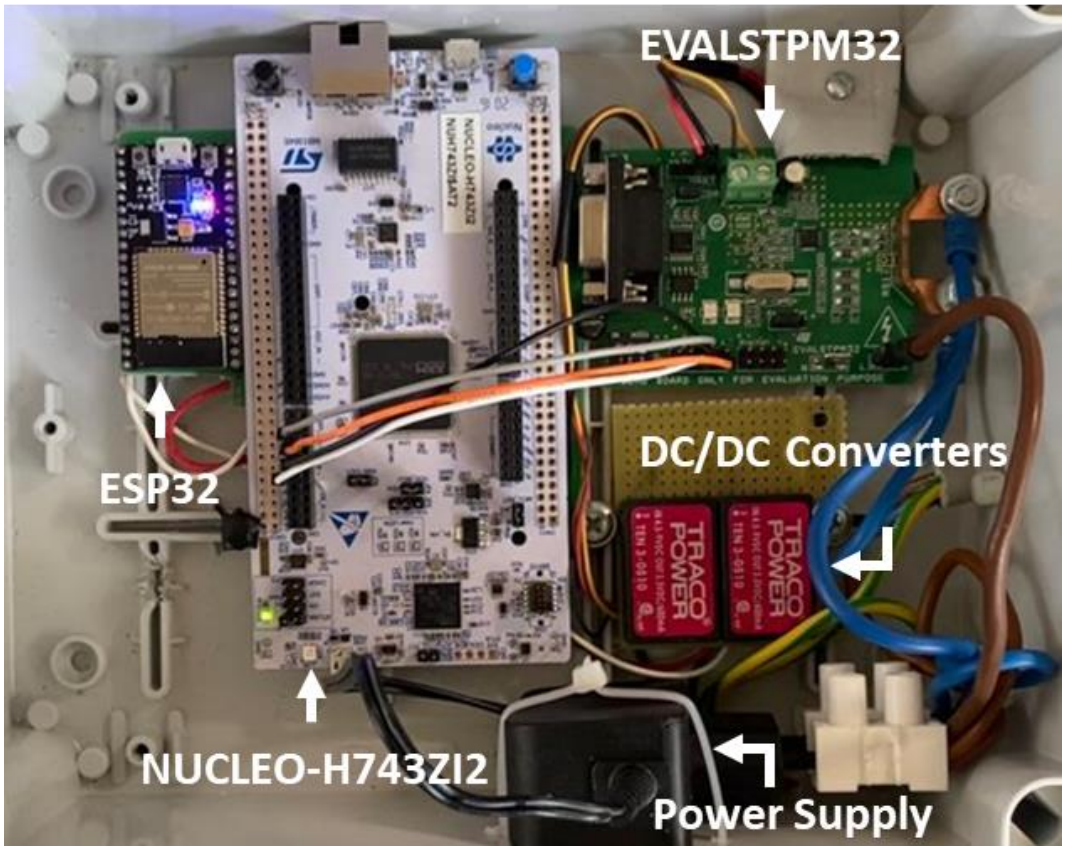

Figure 6.8: The implemented NILM system.

As schematized in Fig. 6.4, each appliance-level power meter consists of one EVALSTPM32 measurement unit, connected to the WLAN via a USR-W610 converter. The USR-W610 is a serial to Wi-Fi and Ethernet converter capable of a bidirectional transparent transmission between RS-232 / RS-485 and Ethernet / Wi-Fi. It allows to assign work details, implement transparent transmission of serial data and TCP/IP data packet via converter.

The USR-W610 supports two wireless interface:

- AP: Access point, central node of a wireless network. In general, a wireless router is an AP, through which other wireless terminals can connect to each other.
- STA: Station, terminal of a wireless network, such as PDA or mobile phone.

Moreover, the USR-W610 supports six operating modes:

- transparent transmission mode,
- serial port command mode,
- HTTP Client mode,
- Modbus TCP to Modbus RTU mode,
- Modbus RTU to Modbus TCP mode,
- AT command mode.

The USR-W610 can open a TCP socket as server or client. To better implement the proposed WLAN network architecture schematized in Fig. 6.4, each wireless module has been set for receiving TCP message through the TCP socket server side. The TCP data are then converted by the wireless module for the RS-232 interface. The EVALSTPM32 board adopts a request-response serial communication handshake so, the bidirectional TCP server socket connection can address the requirement of the communication protocol. The modules are addressed by a static IP address stored in the USR-W610 network configuration, to ensure a point-to-point communication. The Fig. 6.9 shows one of the appliance-level power meters.

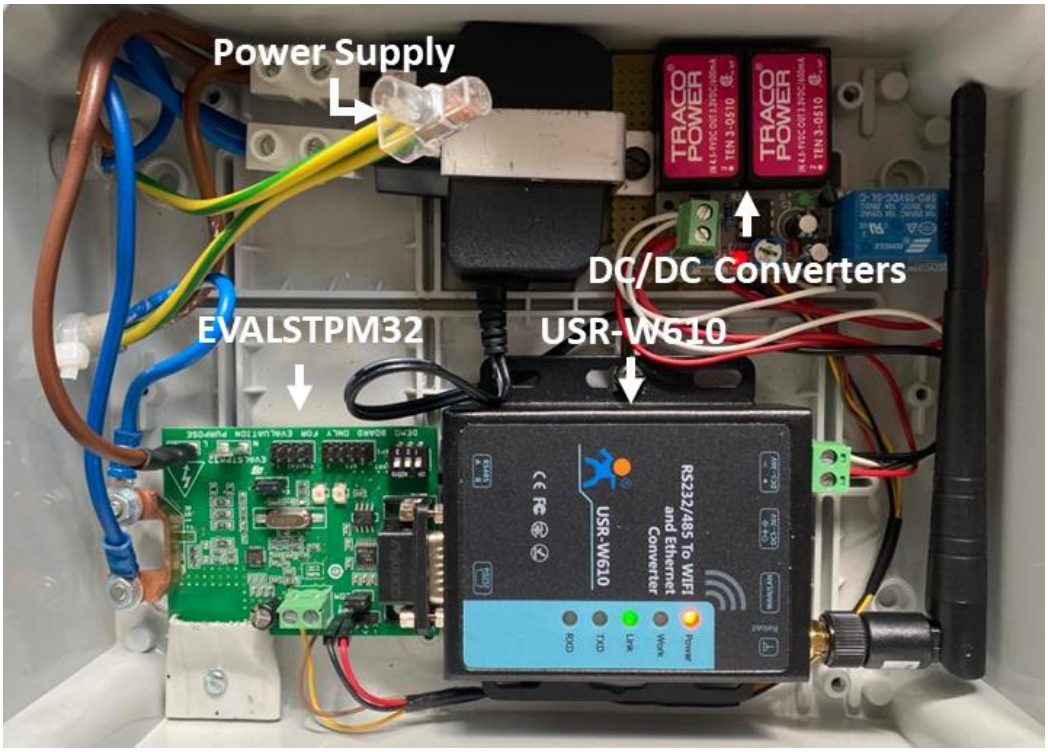

Figure 6.9: One of the appliance-level power meters.

6.2.2 Central concentrator and web server

The main part of the whole system is the Intel NUC NUC5i7RYH, which provides management of the nodes connected to the network, downloading measurement data and storing them on a MySQL database. The system is based on 5th generation Intel Core i7-5557U processor (3.1 GHz up to 3.4 GHz Turbo Dual core, 4 MB Cache, 28W TDP). The Intel NUC supports Intel Hyper-Threading Technology and with 16 GB – DDR3 memory a Windows 7 OS has been used to manage the system.

The application program has been written in Python [185], an object-oriented programming language suitable, among other uses, for developing distributed applications, scripting, numerical computing and system testing.

The tasks performed by the main program are as follows:

- establish the connection with the NILM system and all the appliance-level power meters;
- send the read request to each node connected to the network and receive the data;
- store the data in a MySQL database.

The access to the power meter is done through the multi-threading approach: the python script manages a number of threads equal to the number of the power meter, to ensure that all communication ends in the sampling time (8s). At the end, all acquired data are stored in a MySQL record.

The timestamp is synchronized using the NTP protocol. The Fig. 6.10 shows a flowchart of the main program.

A web server allows supervision of the data during the monitoring period. The interface is developed in Node-RED [195], a programming tool for wiring together hardware devices, APIs and online services. This user interface shows, for each monitored load, 4 different graphs for rms voltage, rms current, active power and reactive power. The graphs show a time window corresponding to the last hour of acquisition, therefore 450 points considering the sampling frequency of 1/8 Hz, comparing the aggregate quantities in each of them with the quantities at the household appliance level. The data is taken directly from the MySQL database. Figs. 6.11 and 6.12 show the web server and the overall installed system, respectively.

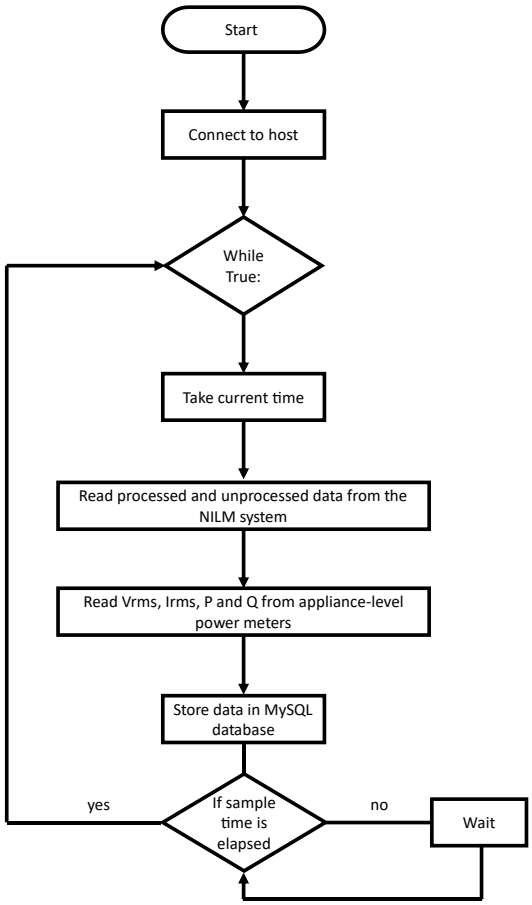

Figure 6.10: The main program.

Figure 6.11: The Web Server.

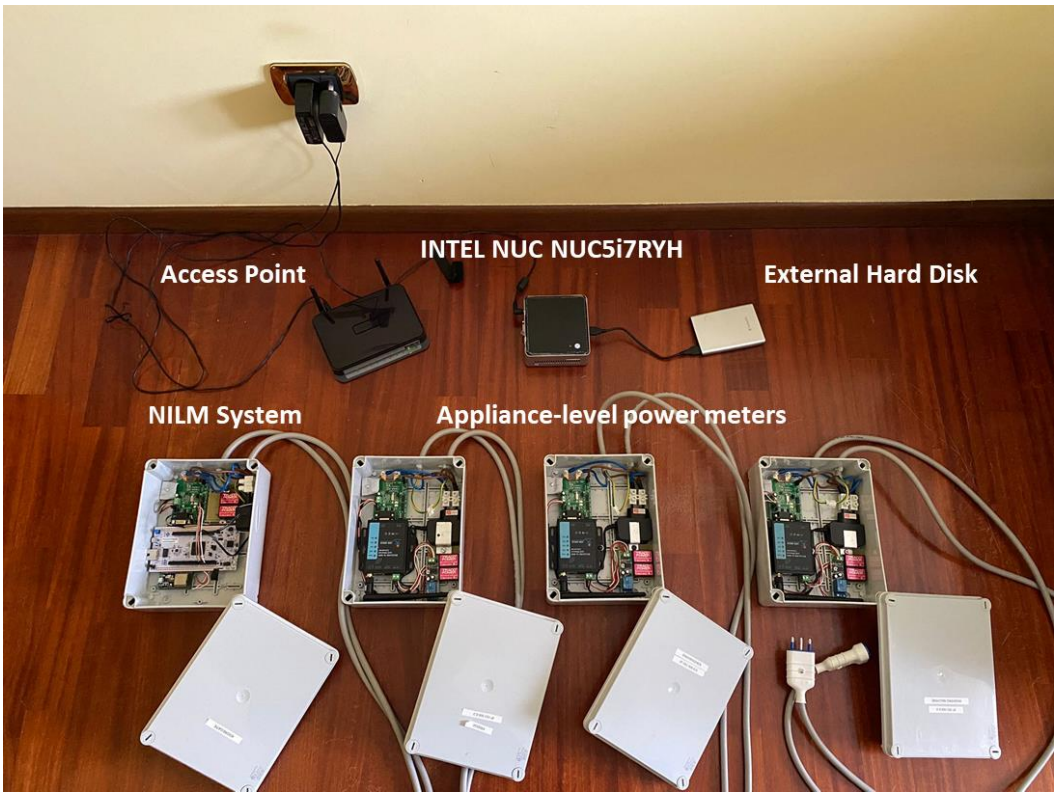

Figure 6.12: The overall installed system.

6.2.3 Calibration of the measurement unit

To ensure that the measured values are metrologically valid, a series of measurements were previously performed on the power meters. A Fluke 6100A Electrical Power Standard was initially used for the verification process. This is a highly accurate voltage and current standard, that provides a reference for measuring electrical power and energy. Once the power meters were verified, more detailed measurements were conducted using a 1-phase measuring system for harmonics and flicker, the HARMONICS-1000. This system allows for simultaneous generation of load voltage and current, allowing for more detailed assessment of the accuracy and reliability of power meters under different operating conditions.

The procedure used in the following calibration is based on an approach presented in detail in the ISO publication "Guide to the Expression of Uncertainty in Measurement (GUM)" [196].

A calibration provides the transducer response as a function of the applied quantity in form (6.4), deriving the K_i coefficients from a least-squares fit to the calibration data.

$$R = K_0 + \sum_{i=1}^{N} K_i V^i \tag{6.4}$$

where R is the transducer response, V is the applied quantity, K_i are the coefficients characterizing the transducer and N is the polynomial order of the calibration function. A polynomial order of 1 was chosen for this work, which is equivalent to a linear interpolation. To conduct the measurements, a variable resistor of known magnitude was used as the load. Data was acquired over a range of resistances, from 23 Ω to 136 Ω, and the load was powered at voltages ranging from 220 V to 240 V. In total, voltage, current, and active power were acquired at 15 different points. Fig. 6.13 shows the calibration setup used in the measurement process. The response of the transducer R_j was measured for each of the different input voltages V_j, currents A_j, and active powers P_j. The K_i coefficients in (6.4) are calculated from a least-squares fit of the sets of measurements $\left(V_j, R_j^V\right)$, $\left(A_j, R_j^A\right)$, $\left(P_j, R_j^P\right)$.

The uncertainty associated with the variation of the measured data from the fitted curve is represented by the standard deviation u_r in (6.5).

$$u_r{}^2 = \frac{\sum_{j=1}^{N}\left(d_j{}^2\right)}{(n-m)} \tag{6.5}$$

where d_j are the differences between the transducer inputs and the responses calculated by (6.4), n is the number of individual measurements in the calibration measurement set, and m is the order of the polynomial plus one.

In order to account for measurement variability, the coefficient of variance c_V was calculated according to (6.6).

$$c_V = \frac{u_r}{\mu} \tag{6.6}$$

where μ represents the mean of the set of measurements. The results of the calibration process indicated that the coefficient of variance obtained for voltage, current, and active power were 0.11%, 0.13%, and 0.87%, respectively.

To further ensure the reliability and accuracy of the measurement system, a characterization of the noise present on the input channels was conducted. Intrinsic

noise in the acquisition and measurement system can negatively impact the identification of loads and compromise the accuracy of the measurements. The results of the noise characterization indicated that the level and type of noise present on the input channels were below the resolution of the measuring system ADC. This means that the noise was not a significant factor in the measurements and did not affect the accuracy of the results obtained.

Overall, these calibration and noise characterization procedures ensured the reliability and accuracy of the measured values and provided a solid foundation for the subsequent analysis and interpretation of the data.

Figure 6.13: The calibration setup.

6.3 Experimental results

As part of the development phase of proposed system, the implemented prototype has been tested, in order to evaluate its performance in real-world scenarios. To achieve this, the NILM system, along with appliance-level power meters, were installed in two houses located in central Italy - one in Marche and the other in Abruzzo.

Over the course of 12 months, from February 2022 to January 2023, the system collected, processed and archived the data of each of the two houses, in each for six consecutive months. This extensive testing phase ensured that the system's performance was thoroughly analyzed and its effectiveness verified under various operating conditions.

In short, as previously discussed, the system is designed to acquire data at a sampling rate of 1/8 Hz. The DL algorithm, implemented in the NUCLEO-H743ZI2 board, processes time windows of 599 points, which is equivalent to 4792 seconds or approximately 80 minutes. The algorithm starts processing when a new active power measurement is available, i.e. every 8 seconds.

An important feature of the implemented system is that it can operate without requiring any prior knowledge of the appliances it is going to monitor. In order to evaluate the performance of the system, its ability to recognize the absorption patterns of the appliances under examination was firstly analyzed, starting from the aggregate energy consumption. To achieve this goal, various load profiles of the appliances were considered, and the responses provided by the system were examined.

The obtained results, plotted in Figs. 6.14 and 6.15, show the four different consumption patterns of monitored dishwashers and washing machines against the acquisition time. The consumption patterns are dependent on the various work cycles set for the appliances, which were taken into account during the analysis, providing valuable insights into the system's performance.

For fridges, the same cycle considerations made for dishwashers and washing machines cannot be applied. Instead, daily consumption was analyzed, as presented in Fig. 6.16, which illustrates four different examples of daily consumption patterns. The Fig. 6.16 provides crucial information on the system's ability to accurately recognize the daily consumption patterns of fridges. The left and right graphs in Fig. 6.16 refer to the first and second houses, respectively. In particular, the two fridges have different consumption patterns, but the DL algorithm manages to adapt

without prior knowledge of the appliance, demonstrating the great flexibility and adaptability of the algorithm.

These figures provide a visual representation of the system's ability to accurately recognize distinct consumption patterns of different appliances. From their analysis, the accuracy and reliability of the system is qualitatively evident.

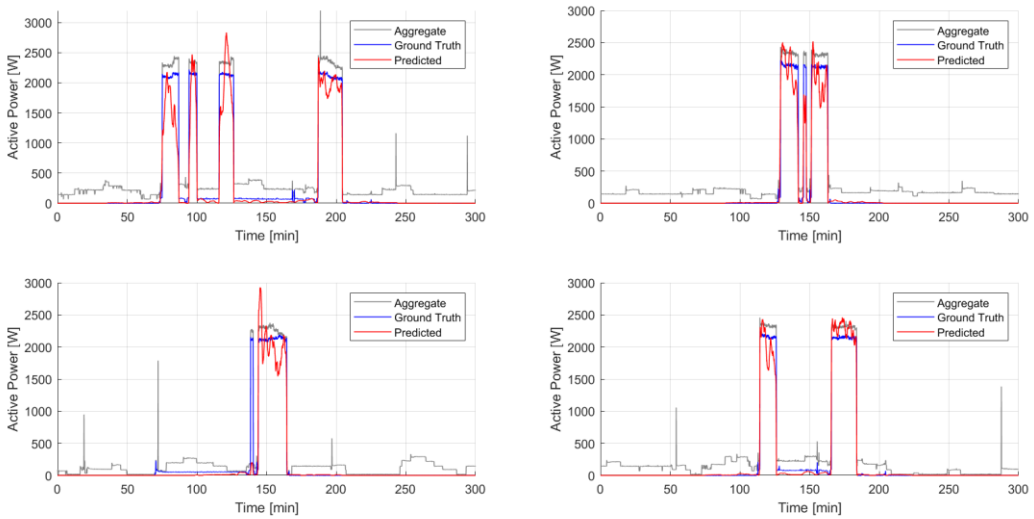

Figure 6.14: Load profiles of dishwashers for different work cycles and corresponding disaggregation result.

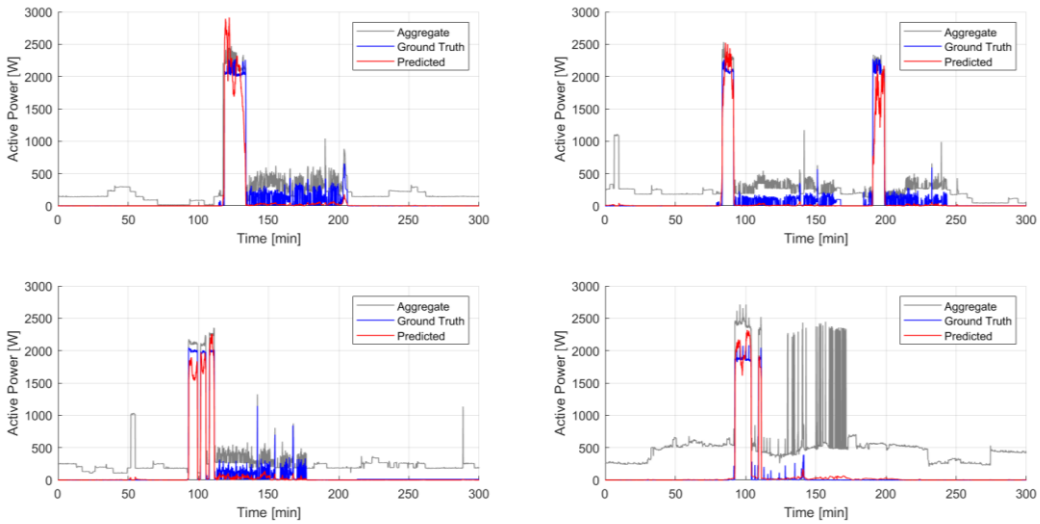

Figure 6.15: Load profiles of washing machines for different work cycles and corresponding disaggregation result.

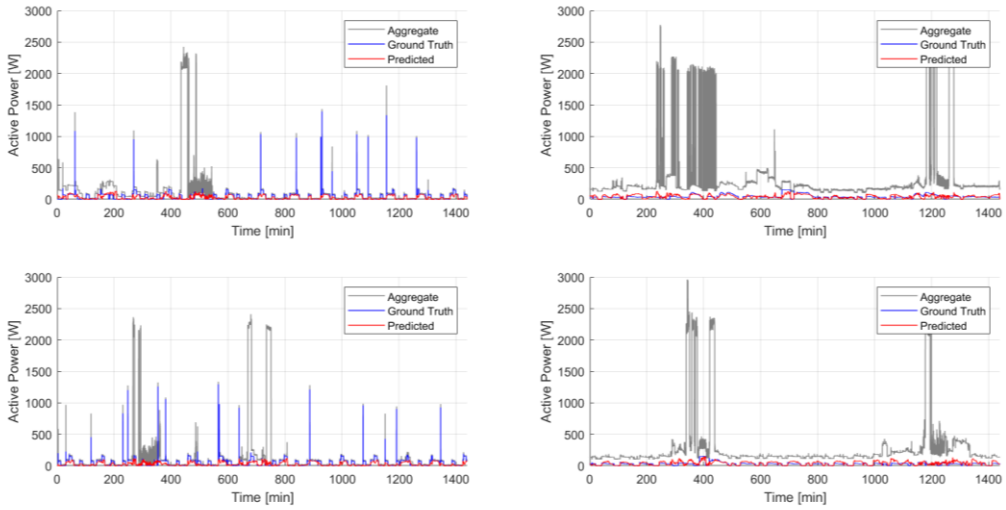

Figure 6.16: Daily consumption patterns of fridges and corresponding disaggregation results.

To also provide a quantitative evaluation of performance, the accuracy of the system in estimating the energy consumption of the appliances in each work cycle was determined, using the metrics typically adopted for NILM systems.

The percentage relative error in estimating energy consumption, often referred to as signal aggregate error (SAE) in the NILM literature, was computed using the (6.7).

$$e\% = \frac{\hat{E}_c - E_c}{E_c} \cdot 100 \tag{6.7}$$

where $\hat{E}_c$ indicates the estimated energy consumption per work cycle and E_c indicates actual energy consumption per work cycle as measured by appliance-level power meters. For each appliance, 50 different work cycles were considered, 25 for each house. Figs. 6.17, 6.18 and 6.19 show the actual and estimated energy for each work cycle (top) and the corresponding relative error (bottom) for dishwashers, washing machines and fridges, respectively. For the same reasons explained above, in the case of fridges, daily consumption will be considered as work cycles.

To summarize the results, Table 6.2 presents the absolute and percentage relative errors in estimating the total energy consumed in the 50 work cycles (50 days in the case of fridges). These metrics offer a comprehensive evaluation of the system's accuracy in estimating energy consumption.

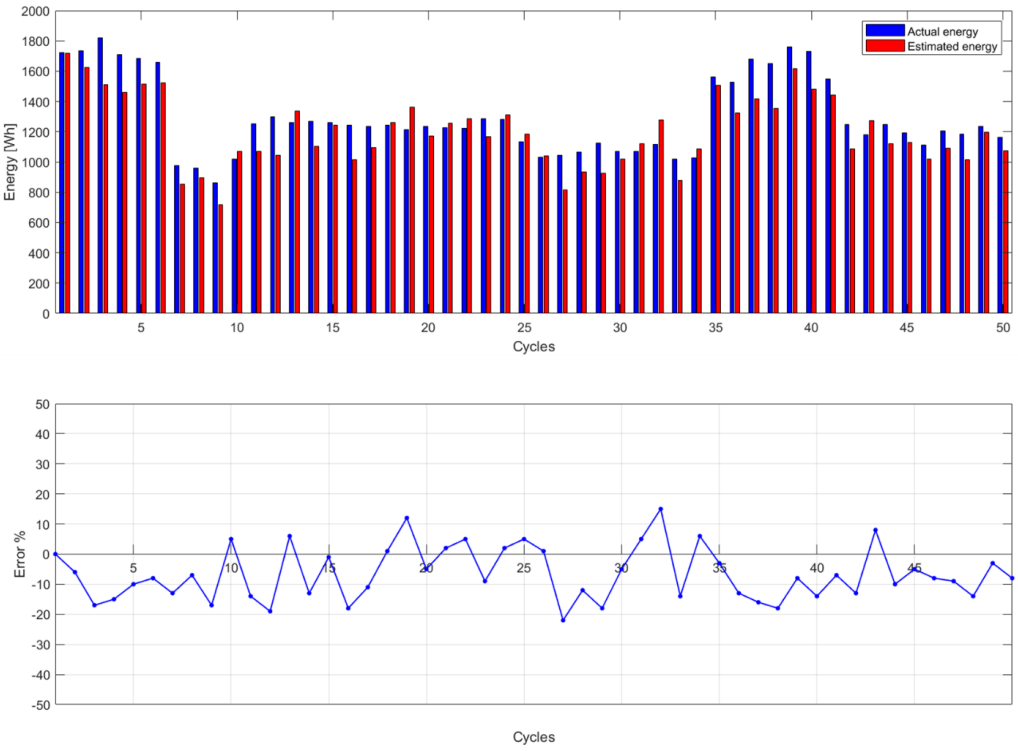

Figure 6.17: Actual and estimated energy per dishwasher work cycle (top) and the corresponding relative error (bottom).

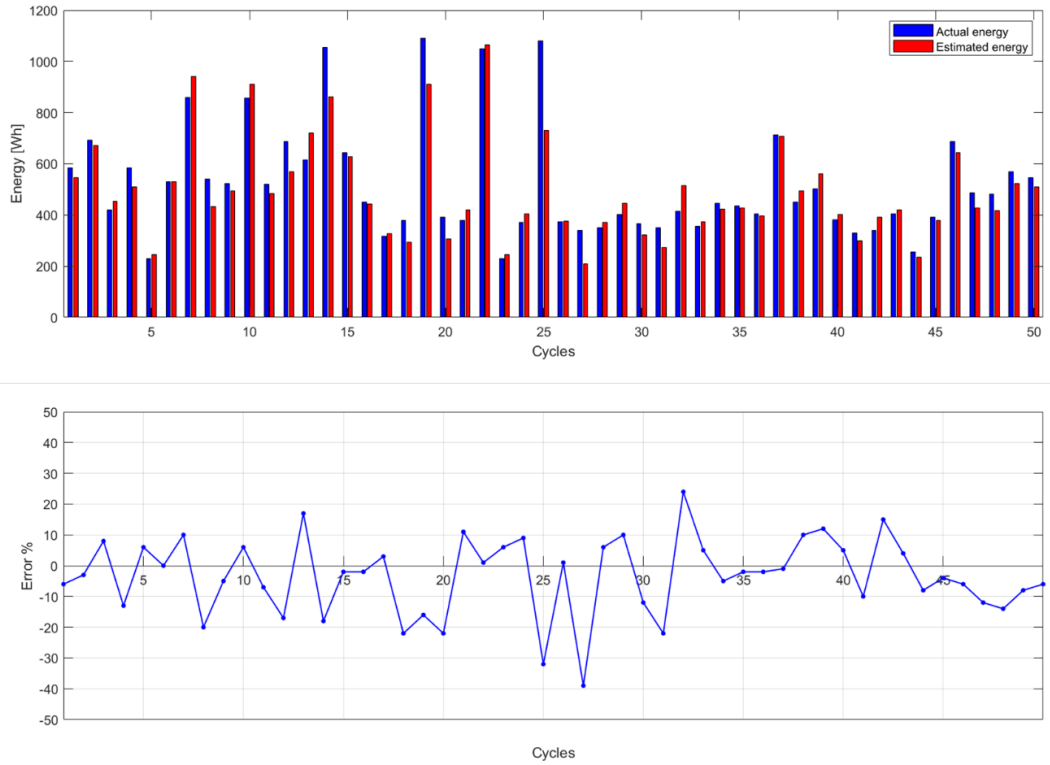

Figure 6.18: Actual and estimated energy per washing machine work cycle (top) and the corresponding relative error (bottom).

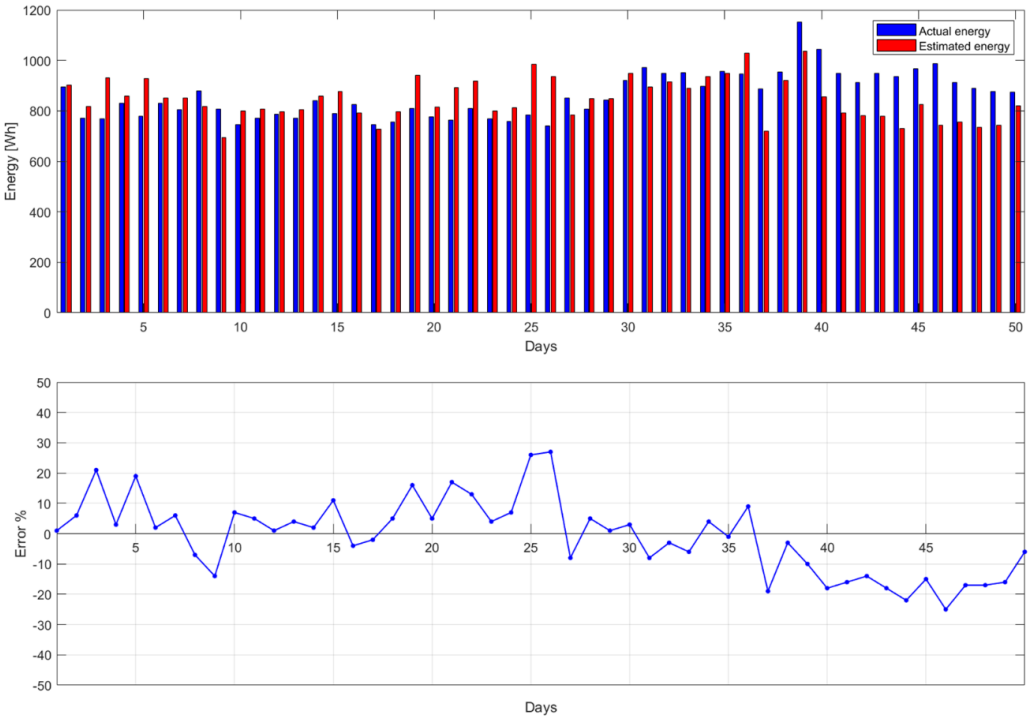

Figure 6.19: Actual and estimated energy per day from fridges (top) and the corresponding relative error (bottom).

Table 6.2: Absolute and percentage relative errors in estimating total energy consumption.

	Absolute error	**Percentage relative error**
Dishwasher	-4558,076 Wh	-7%
Washing machine	-1166,067 Wh	-5%
Fridge	-687,756 Wh	-2%

The analysis of the percentage relative error in (6.7) was extended to a half-yearly basis. To provide a comparison with the existing literature, an additional metric was also considered. The mean absolute error (MAE) was computed to evaluate the accuracy of the model's predictions. The MAE, reported in (6.8), measures the average absolute difference between the actual and estimated active power at each time step.

$$MAE = \frac{1}{T}\sum_{t=1}^{T}|\hat{p}_t - p_t| \tag{6.8}$$

where $\hat{p}_t$ is the estimated power value at time step t, p_t is the measured power value at time step t, and T is the time interval considered. Table 6.3 shows the percentage relative errors and MAEs obtained for the different appliances over the entire dataset for each house (half-year basis).

To better explain the results presented in Table 6.3, Table 6.4 provides additional metrics obtained from a related study [122]. The authors of [122] trained a sequence-to-point DL model on the REFIT dataset [149], as was done in this thesis, but they assessed its generalization capability using the UK-DALE [191] and REDD dataset [88] houses as test sets. These datasets, along with REFIT, were introduced in Section 6.1, and they are widely used to evaluate the performance of energy disaggregation algorithms.

By evaluating the model's ability to generalize across different datasets and households, this study contributes to the understanding of its robustness and applicability to real-world scenarios. Analogously, the results presented in this thesis also demonstrate the generalization capability of the model, even though it was trained on consumption patterns related to UK homes and installed in two Italian homes with different habits and appliances. This indicates that the model can effectively adapt to different settings and perform well under different conditions.

It is worth mentioning that the results presented in [122] were obtained by processing the data offline, which means that the model was trained and tested on pre-recorded data. In contrast, the results presented in this thesis were obtained by implementing the measurement system in real installations, where the model was applied to real-time acquired data. This further confirms the effectiveness and reliability of the model also for on-field applications. Overall, these findings provide important insights into the model's generalization ability, applicability, and scalability, which are critical factors for its practical implementation and adoption.

Table 6.3: Percentage relative error and mean absolute error on a half-yearly basis.

	House 1		**House 2**	
	e%	**MAE**	**e%**	**MAE**
Dishwasher	9%	0.38	11%	0.54
Washing machine	11%	0.57	12%	0.67
Fridge	7%	2.43	10%	1.47

Table 6.4: Percentage relative error and mean absolute error achieved in [122].

	Trained on REFIT and tested on UK-DALE		**Trained on REFIT and tested on REDD**	
	e%	**MAE**	**e%**	**MAE**
Dishwasher	13%	16.49	71%	29.68
Washing machine	50%	14.84	74%	36.83
Fridge	9%	17.00	2%	38.63

Chapter 7

A NILM system based on the SFRA technique

This chapter deals with the problem of identifying the home appliances connected to the network, presenting a measurement system based on the injection of a sinusoidal signal of reduced amplitude and variable frequency, and on the analysis of the responses detected. The system can be classified as non-intrusive monitoring, as it does not require modifications to the electrical system and the measurement system is centralized and can be located at the home meter outlet. The technique, known as SFRA (Sweep Frequency Response Analysis), is a widely used technique, but for another type of application, namely for the diagnosis of electric power transformers and also of electric motors.

The proposed solution is very different from the others present in the literature, which involve the analysis of power signals (typically the current) through the different approaches extensively discussed in the previous chapters. These systems have several advantages, but they all have a common weakness point, linked to the fact that the survey of an appliance requires a change of its state. In fact, they generally pay attention to the transients of the absorbed current, which indicate a change in the operating state of the appliance. The measurement of the current in static conditions does not allow the identification of active devices, except in very particular simple cases.

The approach proposed in this section of the thesis allows to identify the connected devices through a measurement carried out in static conditions (it does not require changes in the state of the devices). The SFRA technique makes it possible to detect a sort of univocal signature of the household appliances connected to the grid, independently of the trend of the absorbed current. This approach, as will be illustrated below, allows the typical problems of NILM systems to be overcome in identifying multi-state appliances or appliances with continuous variable load (or in general electrical loads).

As concerning the classical SFRA technique, all the SFRA measurement instruments available on the market can work only on single devices, that are switched off and disconnected from the grid (off-line). The SFRA system proposed in this article can operate on-line [197], thus allowing to extend its operating range to systems for continuous diagnostics on devices while supplied by mains; no functioning interruptions or disconnection operations are needed, as for the standard SFRA apparatuses.

Whereas as for the proposed SFRA NILM system, it is based on a Machine Learning algorithm, the Support Vector Machine (SVM), capable of determining the status of individual household appliances starting from the measurement obtained by the SFRA system. It was installed on a home test system, and acquired and processed the data locally.

Extensive measurements were made in order to verify the operational characteristics. The results obtained from field applications are also included and discussed.

7.1 Frequency response analysis of household appliances

Classical SFRA technique has been successfully proposed to perform diagnostics on the windings of electric machines during the production process [198],[199]. An electric machine can be considered as a complex electrical network of capacitances, inductances and resistors. As shown in Fig. 7.1, the SFRA instrument injects a sinusoidal excitation voltage (typical amplitude is 10 Vpp) with a continuously increasing frequency into one end of the transformer winding, and measures the signal returning from the other end. This test is conducted with the machine disconnected from the power line. More details are reported in the [198]. Comparison of input and output signals generates a frequency response, which can be compared with previous responses taken as a reference. A degradation of the insulating materials or a change in the shape of the windings will result in a change in the RLC components of the equivalent network and, consequently, in the frequency response curve. Faults can therefore be detected by processing correlation indices between different curves.

In the proposed NILM application, shown in Fig. 7.2, the SFRA technique is applied to the electrical system supplied by the mains, in order to obtain a signature that allows to discriminate different power supply conditions of a domestic system. The applied signal and the output signal, between the terminal of the neutral conductor and the ground, are acquired and processed by the system.

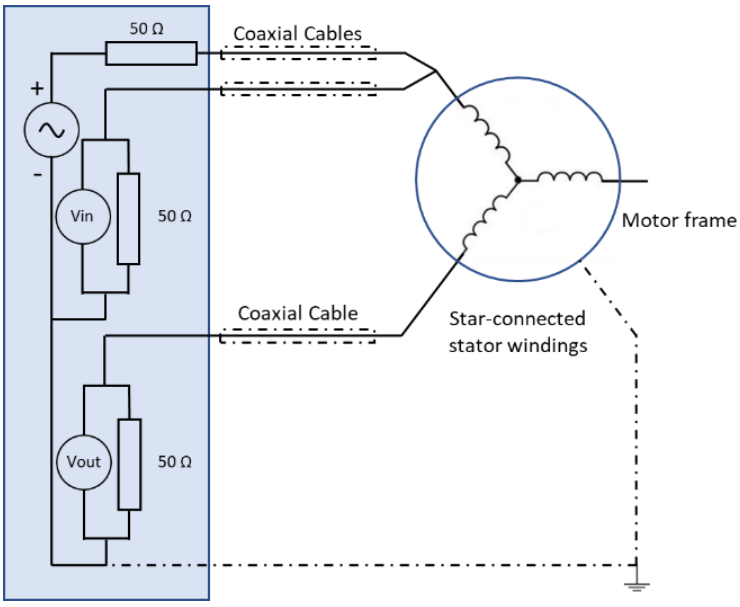

Figure 7.1: SFRA applied to a star connected electric machine.

The proposed measuring system can therefore be easily installed on a normal domestic socket, and this flexibility represents a further innovative advantage.

A low voltage (± 5 Vpp) sinusoidal signal with variable frequency (from 2 kHz to 1.5 MHz) is superimposed on the supply voltage (240 Vrms, 50 Hz) and applied between the power phase conductor terminal and ground.

The signal generator is coupled to the network by means of a band-pass filter, that allows only the passage of the test signal. The two input channels of the measurement circuit are also decoupled from the power supply by two other band-pass filters. The filters block both the fundamental frequency (50 Hz) and the harmonic components (up to 2 kHz) [7].

The frequency response is obtained by injecting a signal, generated at 100 MHz. In order to optimize the memory, the sampling frequency to acquire both applied and output signals is adapted according to the frequency to be analyzed. In detail, the sampling frequency is chosen equal to 25 times the analyzed frequency. To obtain a better resolution and avoid the phenomenon of spectral leakage, the FFT is performed by setting the frequency of one of the bins equal to that of the generated sinusoid. The FFT is also performed on the output signal and the sample in the same container is considered.

A Hanning window with a width equal to the acquisition time (corresponding to 64 cycles of the generated frequency) is used for the processing of the FFT. Downstream of the FFT processing, the system calculates the Vout / Vin ratio. For example, the 1 kHz response is achieved by injecting a 1 kHz sinusoidal signal generated at a frequency of 100 MHz.

The applied signal and the output signal are sampled at a sampling rate of 25 × 1000 = 25000 Hz. A time window of (1 / 1000) × 64 = 0.064 s is considered for the processing of the FFTs, corresponding to 1600 samples. This process is repeated for all the frequencies of interest.

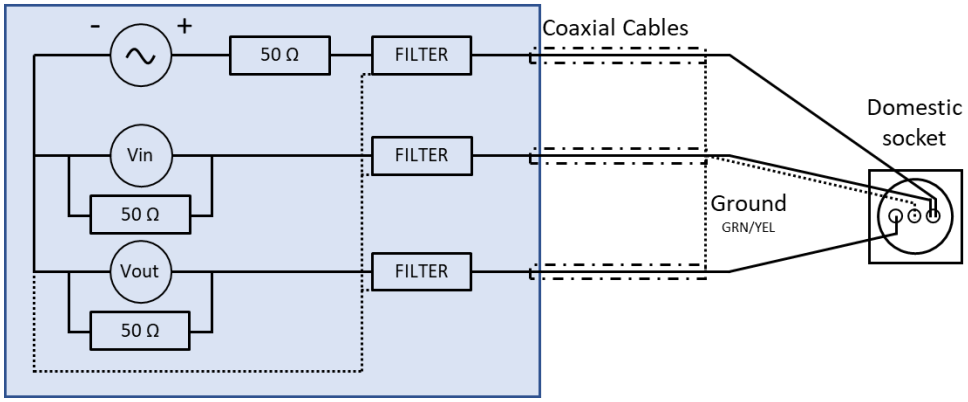

Figure 7.2: SFRA system.

In order to evaluate the validity of the signature for different frequency ranges, 4 sub-bands have been defined:

1) 2-10 kHz;
2) 10-100 kHz;
3) 100 kHz - 1 MHz;
4) 1-1.5 MHz.

For each sub-band, 200 points were initially acquired. These subdivisions were obtained considering the possible response to this type of excitation signal.

The Fig. 7.3 schematically shows the installation of the SFRA system in the test system. From the knowledge coming from the literature about SFRA [7], the low frequency response (2-10 kHz) is characterized by an ohmic-inductive behavior in which the characteristics of the grid upstream of the system are predominant, therefore the contribution of the loads is usually not significant.

More in detail, at low frequencies, the inductance of the power system dominates the electrical behavior. The inductance of the upstream electrical grid, which includes the transformers, cables, and other equipment between the power source and the SFRA system, is typically much larger than the inductance of the downstream system, which includes the loads connected to a domestic, commercial or industrial electrical system.

Therefore, the characteristics of the upstream electrical grid have a greater influence on the SFRA system's low-frequency response than the characteristics of the downstream system. The upstream electrical grid's characteristics, such as its inductance and impedance, determine the overall electrical behavior of the network, and this behavior is reflected in the SFRA system's low-frequency response.

In contrast, the loads connected to the SFRA system typically have a much smaller inductance and impedance, and their impact on the SFRA system's low-frequency response is usually negligible.

The medium frequency response (10 kHz – 1 MHz) is characterized by resonance phenomena. As this band is generally the most interesting in terms of the effect of loads on the response, it has been split into two sub-bands to increase resolution.

The high frequency response (1 - 1.5 MHz) is characterized by capacitive effects, due both to the network and to the user loads and to the connection of the measuring instrument itself, which generally determine a poor reproducibility of the measurement.

The sinusoidal test signal introduces no problems into the system. This is essentially due to the reduced amplitude of the test signal with respect to the line voltage (1.54%), which is fully within the limits imposed by the standard [200].

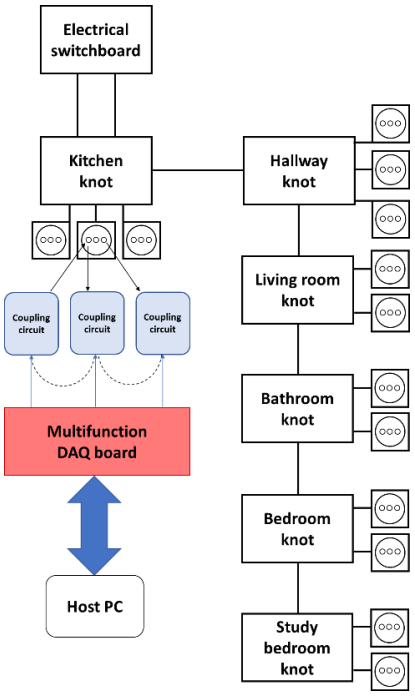

Figure 7.3: Installation of the SFRA in the test system.

During the tests, it has been verified that the signal does not create problems in intelligent automation systems operating with conveyed waves [201]. This is also due to the fact that these systems adopt sophisticated signal modulation algorithms that encode the data transmitted with different sub-carriers, or that widen the transmission band (Spread Spectrum) obtaining a better resistance to interference and noise. Other systems adopt OFDM modulation techniques (Orthogonal Frequency Division Multiplexing) which are even more effective.

Several tests were performed on a residential test facility. A wide variety of loads were taken into consideration, powering them individually or simultaneously and under different working conditions:

1) hairdryer
2) microwave oven
3) lamp
4) laptop
5) induction hob
6) heater
7) drill
8) TV.

Fig. 7.4 shows the frequency response of these appliances when powered individually.

Measurements were made in 24 different power supply scenarios, which were summarized in Table 7.1. Note that scenario 1 represents the case in which none of the appliances is powered (condition indicated with "Open Circuit" in Fig. 7.4). Scenarios 2 to 9 represent the single power supply conditions of household appliances. Scenarios 10 to 24 represent the simultaneous power conditions.

To allow an objective evaluation, Figs. 7.5-7.12 show the lower and upper envelopes of the traces obtained in the presence and absence of each of the 8 considered appliances, obtained following the measurements performed for the different scenarios. Measurements were performed for each of the 24 scenarios reported in Table 7.1, thus obtaining 24 SFRA traces. For each envelope (related to each appliance), the 24 traces were divided into two groups according to the presence or absence of the appliance in the power supply scenario. The envelopes are then obtained by considering the maximum and minimum values of each of the two groups, for each frequency bin.

From these envelopes, it is immediately evident that the contribution of the low frequency measurement (2-10 kHz) is not influenced by the different load configurations, therefore in the rest of the work we will refer only to the other 3 sub-bands. These traces were used as inputs to a machine learning-based classification algorithm, the Support Vector Machine (SVM), in order to determine the correct combination of powered appliances.

A NILM system based on this type of input is easy to install, as it can be connected to a standard domestic socket, such as any household appliance. Traditional NILM systems, on the other hand, measure the aggregate power upstream of the plant and therefore require a more difficult installation.

The obtained measurement represents the transfer function of the equivalent RLC circuit [7]. Therefore, the result is mainly influenced by the physical characteristics of the appliances, rather than by their power absorption. This represents a great advantage for the discrimination of multi-state or continuously variable load appliances (such as drills) whose identification is often critical for systems based on the analysis of power consumption.

The transfer function is minimally influenced by the choice of the socket in which to install the measuring system. Tests have been carried out in all the sockets shown in Fig. 7.3, all the possible positions of the instrument on the various sockets allow the maximum reproducibility of the measurement. Anyway, the instrument is meant to be used on a single socket.

The proposed algorithm is described in Section 7.2.

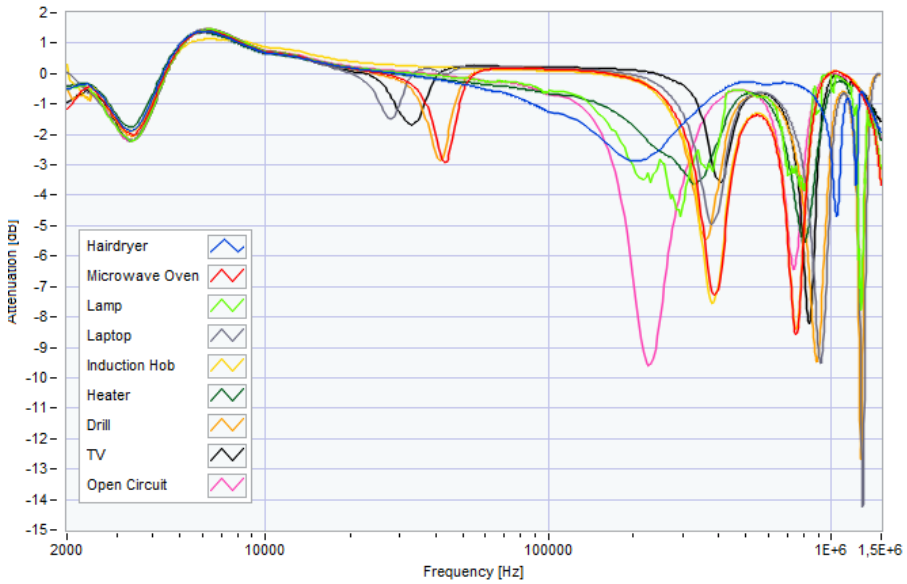

Figure 7.4: Frequency response of individually powered household appliances.

Table 7.1: Power supply scenarios.

	Hairdryer	Microwave oven	Lamp	Laptop	Induction hob	Heater	Drill	TV
1								
2	x							
3		x						
4			x					
5				x				
6					x			
7						x		
8							x	
9								x
10			x	x				
11	x					x		
12		x			x			
13			x	x				x
14	x					x		x
15		x			x			x
16			x	x			x	
17	x					x	x	
18		x			x		x	
19	x		x	x		x		
20		x	x	x	x			
21				x			x	x
22			x	x			x	x
23	x					x	x	x
24		x			x		x	x

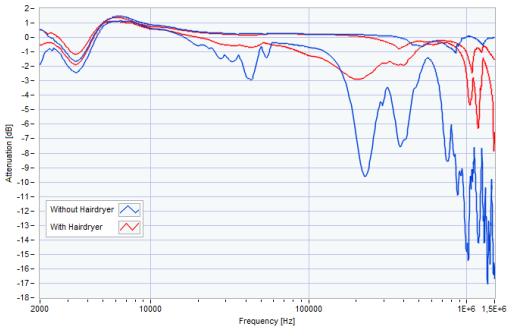

Figure 7.5: Envelopes of the traces obtained in the presence and absence of the hairdryer.

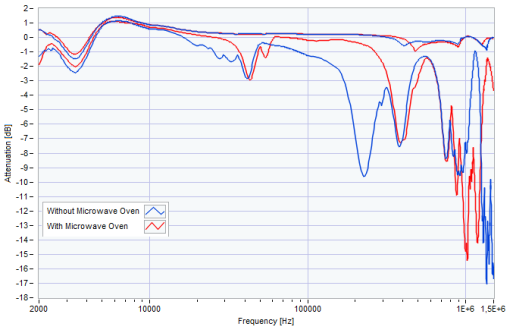

Figure 7.6: Envelopes of the traces obtained in the presence and absence of the microwave oven.

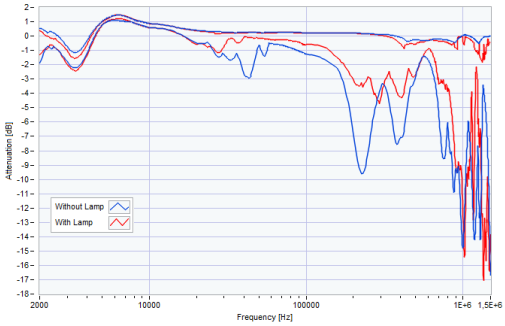

Figure 7.7: Envelopes of the traces obtained in the presence and absence of the lamp.

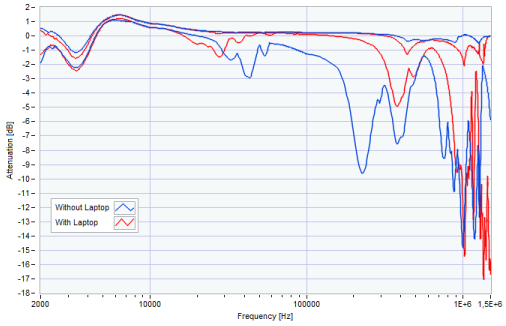

Figure 7.8: Envelopes of the traces obtained in the presence and absence of the laptop.

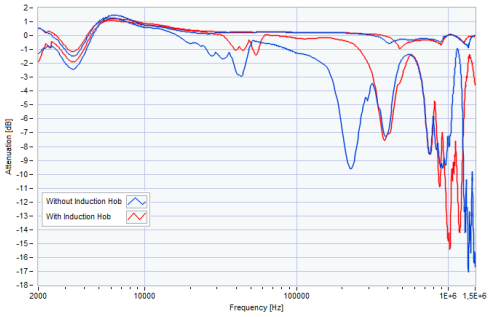

Figure 7.9: Envelopes of the traces obtained in the presence and absence of the induction hob.

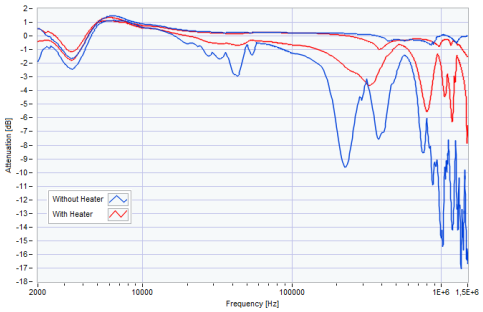

Figure 7.10: Envelopes of the traces obtained in the presence and absence of the heater.

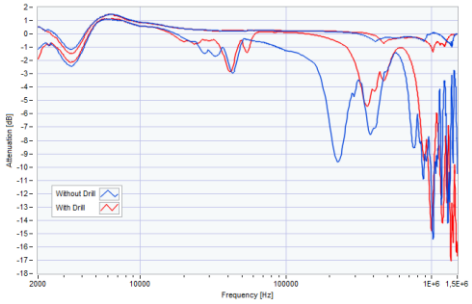

Figure 7.11: Envelopes of the traces obtained in the presence and absence of the drill.

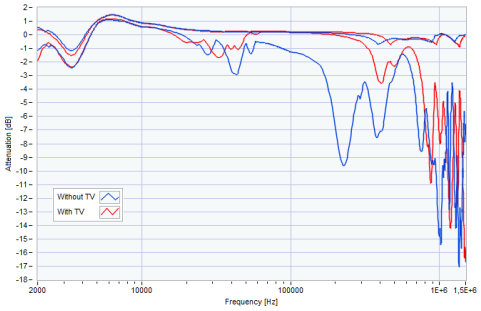

***Figure 7.12:* Envelopes of the traces obtained in the presence and absence of the TV.**

7.2 Machine Learning system

Machine Learning algorithms can be classified into: supervised learning, unsupervised learning, semi-supervised learning, and reinforcement learning. This classification is made in relation to the quantity of data available during the training phase and the type of supervision during the training.

Specifically, in supervised learning, the training data provided to the algorithm includes the desired solutions, called labels. Supervised learning is used to solve two types of problems: classification and regression.

Classification is the problem of cataloging data into two or more classes, so by providing input to the machine learning system, it must return its class of belonging.

On the other hand, regression interpolates data to associate two or more features with each other. By providing the algorithm an input feature, the regressor returns the other feature. A system of estimating the price of houses starting from features such as size, number of rooms and area is a regression system.

The most popular supervised learning algorithms are: k-Nearest Neighbors, Linear regression, Logistic regression, Support Vector Machine (SVM), Decision Trees, Random Forests and Neural Networks.

The NILM problem can be set up either as a regression problem, for example when the algorithm is called to estimate the power absorbed by the single appliance starting from the aggregate power measurement [110], or as a classification problem [10], as in the case in which a starting from the aggregate power measurement it is necessary to determine which appliances are powered and which are not.

The system proposed in this section solves a multi-label classification problem, since starting from an SFRA trace it is possible to identify several powered appliances at the same time. The algorithm used is the SVM, the system configuration and its operation are illustrated in the following paragraphs.

7.2.1 Support Vector Machine

A SVM is one of the most popular models in Machine Learning, as it is very powerful and versatile [104]. SVMs are best suited for classifying complex but small to medium-sized datasets.

While the classic classification algorithms discriminate on the basis of characteristics common to each class, the SVM algorithms build the model on the basis of the most difficult samples to discriminate, i.e. the most similar samples but belonging to different classes.

In this sense, the only samples used in the construction of the model are called support vectors. The other samples are therefore useless.

Based on the support vectors, the algorithm finds the optimal hyperplane that separates them, and which can then be used to discriminate new samples. In other words, adding more formation samples far from the hyperplane (therefore not particularly complex to classify) will not affect the decision boundary at all, which will be completely determined by the samples located at the edge of the hyperplane.

Consider a case in which the samples to be classified are defined by only two features.

This case can be represented on a two-dimensional plane, as shown in Fig. 7.13. An SVM algorithm looks for the line capable of maximizing the margin between the most similar samples belonging to different classes, i.e. the support vectors.

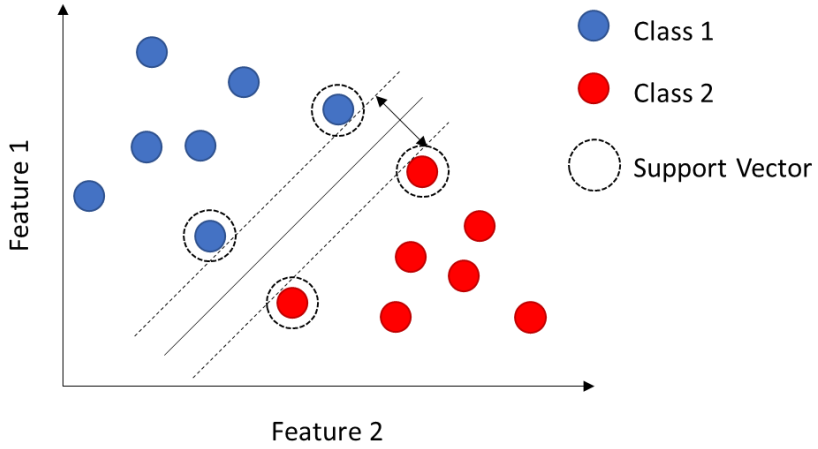

Figure 7.13: Representation of a linear classification problem in which the samples are defined by only two features.

However, not all classification problems are linear, in fact in some cases it is not possible to separate the classes with a straight line, therefore we speak of non-linear classification. The kernel trick [202] is used to solve non-linear classification problems with SVM algorithms. More in detail, a polynomial kernel was used to determine the presence or not of an appliance starting from the SFRA traces.

Using a polynomial kernel means determining similarity not only by processing the features of the input samples, but also their combinations, as shown in Fig. 7.14.

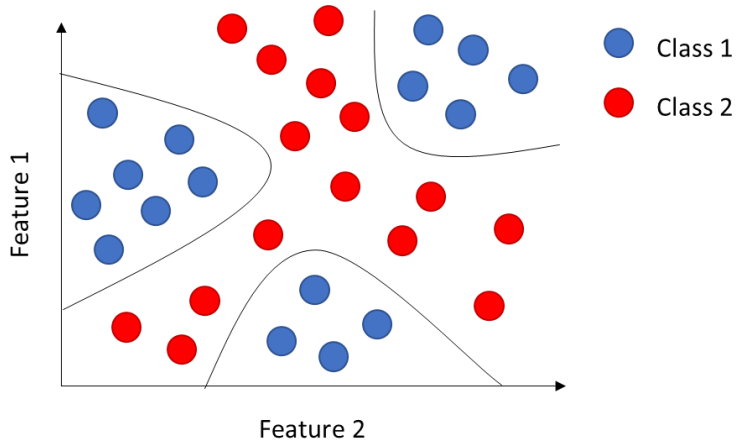

Figure 7.14: Representation of a non-linear classification problem in which the examples are defined by only two features.

7.2.2 The proposed structure

The previous paragraph refers to a two-dimensional problem. However, in the proposed system, the input is the trace obtained from the SFRA system, so each point of the trace represents a feature for the SVM. The algorithm must have a number of input functions equal to the number of bins of the measured frequency response.

The problem is also attributable to a multi-label classification problem, where a single sample can belong to multiple defined classes, unlike what happens in a multi-class classification where each sample can uniquely belong to only one class.

In fact, the purpose of the system is to determine the status (ON or OFF) of the appliances. This means that the number of classes is equal to that of the appliances and the belonging of an SFRA trace to a certain class will be indicative of the ON state of that appliance. A single SFRA trace must therefore be able to be associated with multiple classes (or labels) as the system must be able to recognize the loads even under simultaneous power supply conditions.

SVMs are not natively capable of performing multi-class or multi-label classifications since, as explained above, an SVM defines a hyperplane that separates classes equidistantly in order to guarantee the maximum margin. When the number of classes rises to three or more, thus passing from a binary classification to a multiclass, it is possible to guarantee equidistance only between two of the classes, discarding this property with all the other classes.

To solve this classification problem, which involves assigning multiple labels to an instance, we converted it to multiple binary classification problems. An SVM was therefore associated with each household appliance, which performs a binary

classification in order to determine its ON or OFF status, starting from the SFRA trace.

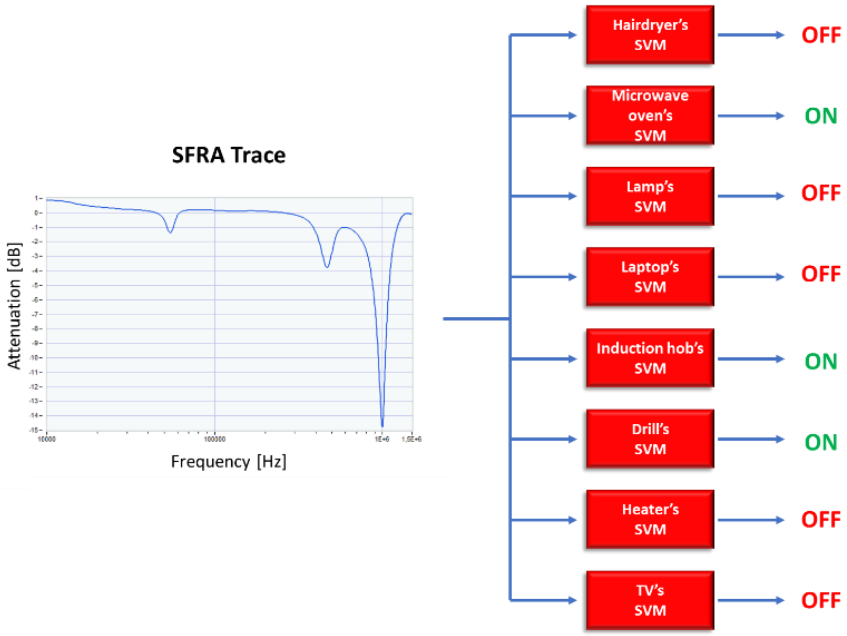

Figure 7.15: The proposed structure.

7.3 Experimental results

As part of the development phase, the proposed algorithm was implemented and tested to evaluate its performance with real data.

7.3.1 Proposed system setup

As explained in Section 7.1, the SFRA technique was performed by plugging the instrument into a standard household socket. As previously discussed, the input signal is a variable frequency sinusoidal signal applied between the phase conductor terminal and ground, while the output signal is the measured signal between the neutral conductor terminal and ground. Both signals are acquired and processed. Fig. 7.16 shows the measurement system used.

The measurement system must be connected to the test system by means of cables with suitable bandwidth, and the same characteristic impedance of the generator to avoid reflection and signal mismatch, and to improve the sensitivity, repeatability and reliability of the measurement.

The input signal and related acquisition for the SFRA were performed using the Digilent Analog Discovery 2 NI Edition card with BNC adapter.

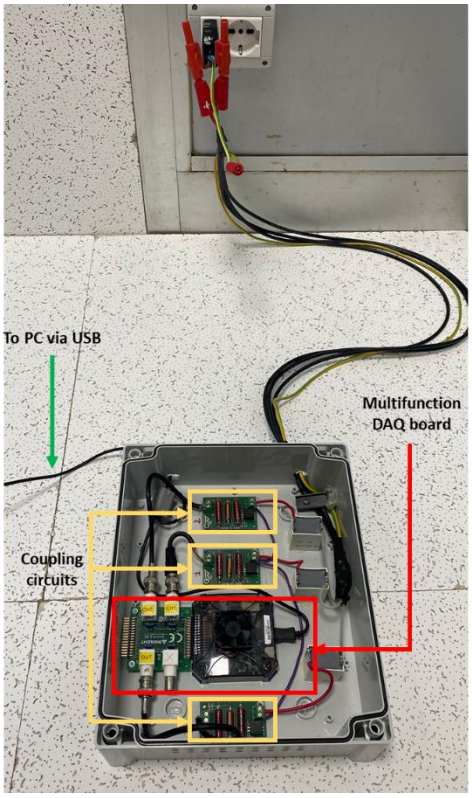

Figure 7.16: The implemented SFRA measurement system.

The control system was developed using LabVIEW and run on a PC, this software automatically programs the Discovery FPGA at startup, with a configuration file designed to implement the measurement application. Once programmed, the integrated FPGA communicates with the PC via a USB 2.0 connection. The PC allows to create the user interface to access the data and process them in the experimental phase. A final NILM system can bypass the PC by integrating post-processing directly on the system.

The Discovery has a ± 25 V input range, a 14-bit resolution, a 100 MHz sampling frequency and a 30 MHz bandwidth. It is equipped with an arbitrary function generator with an output range of ± 5 V, bandwidth of 20 MHz and sampling rate of 100 MHz.

In order to avoid unwanted overvoltages due to resonance phenomena at high frequencies, the amplitude of the applied signal must not exceed a few volts (5 Vpp in the present case). The accuracy of the adopted measurement system, as discussed in a previous paper [203], has been evaluated using a reference parallel LCR circuit. This circuit consists of a 50 **Ω** resistive adapter, a fixed inductance and a variable capacitance. The referenced values of the circuit impedance were measured with a Keysight E4980AL precision LCR meter. The estimated accuracy of Vout/Vin ratio was better than ±0.2 dB, in the interval from +5 to −25 dB, in the frequency range from 5 kHz to 1.5 MHz.

The SVM was implemented on a desktop computer (based on the Windows 10 × 64-bit operating system) using the open-source Python 3.7 from Anaconda [185], the machine learning algorithm was developed using the Scikit-learn library.

7.3.2 The achieved results

The proposed measurement technique is innovative and does not appear to have been tested by other authors. Due to the specificity of the acquired data (frequency response), there are no public datasets used by other authors against which to compare the performance of the proposed algorithm [12].

The measurement system was installed on a test facility, which was designed to generate electrical loads created by domestic users, as part of the “non-intrusive infrastructure for monitoring loads in residential users” research project. The facility, which is located in the Electrical Engineering Laboratory of the University of L’Aquila (I), allows the generation of electrical loads in a single or simultaneous way.

During the test phase, various parameters were evaluated in order to define: the most significant sub-bands, the number of measurement points to be acquired and the number of training examples needed to obtain satisfactory performance.

To this end, precision, recall and F1-Score during classification were evaluated [186]. These parameters have already been defined and used in Chapter 5, but for ease of reading they are given below:

$$Precision = \frac{TP}{TP + FP} \tag{7.1}$$

$$Recall = \frac{TP}{TP + FN} \tag{7.2}$$

$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{7.3}$$

The concept of positive has been attributed to the ON state of household appliances and that of negative to the OFF state. Precision indicates, of all the times the system has provided an indication of the ON state of an appliance, how many times the prediction has been correct. Precision does not take FNs into account. On the other hand, Recall indicates how many times the system has provided a correct indication about the ON state of the appliance compared to all the samples in which the appliance was actually in the ON state. Recall does not take FPs into account. To have a metric capable of taking into account both FPs and FNs, the F1-Score is used, that is a harmonic mean of Precision and Recall.

Since, as already explained above, each appliance is associated with an SVM algorithm that reveals its presence or not, the performance of each SVM has been evaluated individually.

We started by acquiring 20 samples for each of the 24 scenarios, for a total of 480 training samples. Each sample consists of an SFRA trace in which 200 points were acquired for each of the 3 sub-bands. Performance was evaluated on a test set consisting of 50 samples for each scenario, for a total of 1200 test samples.

The obtained results, shown in Table 7.2, are already excellent, as 480 training samples are a relatively low number considering that the acquisition of a single sample takes about 40 seconds.

The system does not make mistakes for 5 of the 8 appliances analyzed, however showing high performance also for the other 3 appliances. To define which of the 3 sub-bands made the most significant contribution to the identification of household appliances, the performance of the system was evaluated by providing the 3 sub-bands separately as input to the machine learning system. The results are reported in Table 7.3.

Table 7.2: The results obtained with 480 training samples and 200 points for each sub-band.

	Total Errors	FP	FN	Precision	Recall	F1-Score
Hairdryer	0	0	0	1.00	1.00	1.00
Microwave Oven	0	0	0	1.00	1.00	1.00
Lamp	27	0	27	1.00	0.92	0.96
Laptop	0	0	0	1.00	1.00	1,00
Induction Hob	0	0	0	1.00	1.00	1.00
Heater	5	5	0	0.98	1.00	0.99
Drill	29	29	0	0.93	1.00	0.97
TV	0	0	0	1.00	1.00	1.00

Table 7.3: Performance evaluation for each sub-band.

10-100 kHz

	Total Errors	FP	FN	Precision	Recall	F1-Score
Hairdryer	98	98	0	0.75	1.00	0.86
Microwave Oven	50	0	50	1.00	0.83	0.91
Lamp	110	96	14	0.78	0.96	0.86
Laptop	51	51	0	0.89	1.00	0.94
Induction Hob	0	0	0	1.00	1.00	1.00
Heater	61	61	0	0.83	1.00	0.91
Drill	48	48	0	0.89	1.00	0.94
TV	0	0	0	1.00	1.00	1.00

100 kHz - 1 MHz

	Total Errors	FP	FN	Precision	Recall	F1-Score
Hairdryer	0	0	0	1.00	1.00	1.00
Microwave Oven	0	0	0	1.00	1.00	1.00
Lamp	141	30	111	0.89	0.68	0.77
Laptop	9	0	9	1.00	0.98	0.99
Induction Hob	59	9	50	0.97	0.83	0.89
Heater	5	5	0	0.98	1.00	0.99
Drill	116	106	10	0.79	0.98	0.87
TV	0	0	0	1.00	1.00	1.00

1-1.5 MHz

	Total Errors	FP	FN	Precision	Recall	F1-Score
Hairdryer	2	2	0	0.99	1.00	0.99
Microwave Oven	93	85	8	0.77	0.97	0.86
Lamp	115	0	115	1.00	0.67	0.80
Laptop	71	6	65	0.98	0.84	0.90
Induction Hob	79	74	5	0.80	0.98	0.88
Heater	29	29	0	0.91	1.00	0.95
Drill	90	76	14	0.84	0.97	0.90
TV	48	39	9	0.91	0.98	0.94

In light of these results, it was decided to consider only the sub-bands 10 - 100 kHz and 100 kHz - 1 MHz, in order to reduce the time required for the measurement. This reduces the time it takes to acquire a single trace to 22.56 seconds. Table 7.4 reports the performance evaluation using only the first two sub-bands as input.

Comparing the results with those of Table 7.2, it can be seen that the performance of the system has remained roughly unchanged. There is a significant improvement in the detection of the drill, highlighting that the 1-1.5 MHz sub-band introduced useless randomness for identification purposes. In this way 400 points are acquired in the 10 kHz - 1 MHz frequency band.

The possibility of decreasing the number of acquired points has been evaluated. Therefore, in Table 7.5 the performances obtained for 200, 134 and 100 points are reported. Performance proved to be very good even using only 100 measurement points as system input. In these conditions, in fact, the system made errors only for 3 of the 8 appliances analyzed, while maintaining a minimum F1-Score of 0.94. This reduction allowed a decrease in the execution time of the measurement system from 22.56 seconds to 6.09 seconds.

The performances shown so far always foresee 480 training samples (20 for each of the 24 scenarios). As a final analysis, the impact of the number of training samples on performance was evaluated. Table 7.6 reports the results obtained using an SFRA trace consisting of 100 points acquired in the 10 kHz - 1 MHz frequency band, reducing the number of samples used in the training phase.

The system maintains interesting performances even when trained with only 1 training sample for each scenario (therefore with 24 total training samples). This is mainly because the SVM natively suffers more from the quality of the training samples rather than the quantity, precisely because it builds a model based only on the most difficult samples to discriminate.

Lower performance was found in the detection of Lamp, Laptop and Drill. In the case of the Lamp this is due to the insignificance of its related load, compared to the overall network, while in the case of the Laptop and Drill it is due to the extreme variability of their working conditions. However, F1-Score values of 0.78, 0.87 and 0.94 respectively can be considered largely satisfactory for a trained system with such a small number of samples.

Table 7.4: The results obtained with 480 training samples and 200 points for each sub-band, using only the first two sub-bands.

	Total Errors	FP	FN	Precision	Recall	F1-Score
Hairdryer	0	0	0	1.00	1.00	1.00
Microwave Oven	0	0	0	1.00	1.00	1.00
Lamp	29	0	29	1.00	0.92	0.96
Laptop	4	4	0	0.99	1.00	0.99
Induction Hob	0	0	0	1.00	1.00	1.00
Heater	3	3	0	0.99	1.00	0.99
Drill	7	7	0	0.98	1.00	0.99
TV	0	0	0	1.00	1.00	1.00

Table 7.5: Performance evaluation as the points acquired decrease.

10 kHz – 1 MHz (200 points)						
	Total Errors	**FP**	**FN**	**Precision**	**Recall**	**F1-Score**
Hairdryer	0	0	0	1.00	1.00	1.00
Microwave Oven	0	0	0	1.00	1.00	1.00
Lamp	39	0	39	1.00	0.89	0.94
Laptop	0	0	0	1.00	1.00	1.00
Induction Hob	0	0	0	1.00	1.00	1.00
Heater	2	2	0	0.99	1.00	1.00
Drill	5	5	0	0.99	1.00	0.99
TV	0	0	0	1.00	1.00	1.00
10 kHz – 1 MHz (134 points)						
	Total Errors	**FP**	**FN**	**Precision**	**Recall**	**F1-Score**
Hairdryer	0	0	0	1.00	1.00	1.00
Microwave Oven	0	0	0	1.00	1.00	1.00
Lamp	26	0	26	1.00	0.93	0.96
Laptop	0	0	0	1.00	1.00	1.00
Induction Hob	0	0	0	1.00	1.00	1.00
Heater	5	5	0	0.98	1.00	0.99
Drill	5	5	0	0.99	1.00	0.99
TV	0	0	0	1.00	1.00	1.00
10 kHz – 1 MHz (100 points)						
	Total Errors	**FP**	**FN**	**Precision**	**Recall**	**F1-Score**
Hairdryer	0	0	0	1.00	1.00	1.00
Microwave Oven	0	0	0	1.00	1.00	1.00
Lamp	42	0	42	1.00	0.88	0.94
Laptop	0	0	0	1.00	1.00	1.00
Induction Hob	0	0	0	1.00	1.00	1.00
Heater	6	6	0	0.98	1.00	0.99
Drill	2	2	0	0.99	1.00	0.99
TV	0	0	0	1.00	1.00	1.00

Table 7.6: Performance evaluation as training samples decrease.

10 kHz - 1 MHz (100 points, 15 samples for each scenario)						
	Total Errors	**FP**	**FN**	**Precision**	**Recall**	**F1-Score**
Hairdryer	0	0	0	1.00	1.00	1.00
Microwave Oven	0	0	0	1.00	1.00	1.00
Lamp	42	0	42	1.00	0.88	0.94
Laptop	0	0	0	1.00	1.00	1.00
Induction Hob	0	0	0	1.00	1.00	1.00
Heater	6	6	0	0.98	1.00	0.99
Drill	3	3	0	0.99	1.00	0.99
TV	0	0	0	1.00	1.00	1.00
10 kHz - 1 MHz (100 points, 10 samples for each scenario)						
	Total Errors	**FP**	**FN**	**Precision**	**Recall**	**F1-Score**
Hairdryer	0	0	0	1.00	1.00	1.00
Microwave Oven	0	0	0	1.00	1.00	1.00
Lamp	59	0	59	1.00	0.83	0.91
Laptop	2	0	2	1.00	0.99	0.99
Induction Hob	0	0	0	1.00	1.00	1.00
Heater	5	5	0	0.98	1.00	0.99
Drill	23	21	2	0.95	1.00	0.97
TV	0	0	0	1.00	1.00	1.00
10 kHz - 1 MHz (100 points, 5 samples for each scenario)						
	Total Errors	**FP**	**FN**	**Precision**	**Recall**	**F1-Score**
Hairdryer	5	5	0	0.98	1.00	0.99
Microwave Oven	0	0	0	1.00	1.00	1.00
Lamp	71	0	71	1.00	0.80	0.89
Laptop	26	0	26	1.00	0.94	0.97
Induction Hob	0	0	0	1.00	1.00	1.00
Heater	16	16	0	0.95	1.00	0.97
Drill	31	29	2	0.93	0.99	0.96
TV	0	0	0	1.00	1.00	1.00
10 kHz - 1 MHz (100 points, 1 samples for each scenario)						
	Total Errors	**FP**	**FN**	**Precision**	**Recall**	**F1-Score**
Hairdryer	5	5	0	0.98	1.00	0.99
Microwave Oven	1	1	0	0.99	1.00	0.99
Lamp	125	0	125	1.00	0.64	0.78
Laptop	95	2	93	0.99	0.77	0.87
Induction Hob	0	0	0	1.00	1.00	1.00
Heater	17	17	0	0.95	1.00	0.97
Drill	53	53	0	0.88	1.00	0.94
TV	8	8	0	0.98	1.00	0.94

In order to provide an overall assessment of the system's performance, metrics widely used for multi-label classification systems were used: micro-average and macro-average.

As reported in (7.4), (7.5) and (7.6), in the micro-average all TPs, TNs, FPs and FNs are summed for all labels and subsequently averaged.

$$Precision_{micro-averaging} = \frac{\sum_{n=1}^{N} TP_n}{\sum_{n=1}^{N} TP_n + FP_n} \tag{7.4}$$

$$Recall_{micro-averaging} = \frac{\sum_{n=1}^{N} TP_n}{\sum_{n=1}^{N} TP_n + FN_n} \tag{7.5}$$

$$F1 - score_{micro-averaging} = \frac{2 \times Precision_{micro-averaging} \times Recall_{micro-averaging}}{Precision_{micro-averaging} + Recall_{micro-averaging}} \tag{7.6}$$

On the other hand, the macro-average, as reported in (7.7), (7.8) and (7.9), is simply the average of Precision and Recall for each label.

$$Precision_{macro-averaging} = \frac{\sum_{n=1}^{N} Precision_n}{N} \tag{7.7}$$

$$Recall_{macro-averaging} = \frac{\sum_{n=1}^{N} Recall_n}{N} \tag{7.8}$$

$$F1 - score_{macro-averaging} = \frac{2 \times Precision_{macro-averaging} \times Recall_{macro-averaging}}{Precision_{macro-averaging} + Recall_{macro-averaging}} \tag{7.9}$$

The difference between the two lies in the fact that the micro-average reflects any imbalances in the dataset. Unbalance means test samples in a greater number of one or more classes than the others. In other words, having more samples for a given scenario, the macro-average, by making a simple average of Precision, Recall and F1-Score, does not consider this imbalance. On the contrary, the micro-average takes these situations into account.

In the case in question, the dataset is balanced; therefore, both averages are functional and adequate for verifying the performance of this system. Table 7.7

reports the micro-averages and macro-averages calculated on the values reported in Table 7.6.

Table 7.7: Impact of the size of the training set on multi-label classification.

	Micro-average			Macro-average		
Training samples for each scenario	**Precision**	**Recall**	**F1-Score**	**Precision**	**Recall**	**F1-Score**
20	0.99	0.98	0.99	0.99	0.98	0.99
15	0.99	0.98	0.99	0.99	0.98	0.99
10	0.99	0.97	0.98	0.99	0.97	0.98
5	0.98	0.96	0.97	0.98	0.96	0.97
1	0.96	0.92	0.94	0.97	0.92	0.94

7.3.3 Analysis of the electromagnetic compatibility

A further consideration must be made regarding the problem of integrating the proposed system into an electrical system. As explained above, there is no interference with the normal functioning of the devices during system operation. Furthermore, the system poses no problems to the EMI filters which are generally placed at the input of many of the monitored devices, since the powers involved, which can be associated with the test signal, are extremely low.

To analyze in detail the operating conditions of the measurement system, it was also simulated in a SPICE environment. Specifically, the simulation was oriented to the analysis of the effects produced by the test signal on commercial EMI filters that could be connected (to other devices) in proximity to the system under test. The analysis was extended to the entire range of frequencies involved; as a reference, a commercial EMI filter family has been considered [204] for standard use in commercial and residential apparatuses for AC currents up to 16 A_{rms}, in single-phase systems.

The analysis was extended to the entire range of frequencies involved. The Fig. 7.17 summarizes the scheme considered for the simulation. The resistance R_{Load} equal to 50 Ω has been chosen in order to simulate the load of a generic household appliance (230 V_{rms} / 50 Ω = 4.6 A_{rms}).

The features of the filters developed for the SFRA apparatus [197] are in Figs. 7.18 and 7.19.

The response of the system was evaluated by varying the frequency in the range in which the proposed system operates in the final configuration (10 kHz - 1 MHz). The frequency response of the current entering the EMI filter was evaluated. Several simulations were carried out by varying the RLC parameters of the EMI filter. The current was found to be harmless across the entire spectrum. As an example, Fig. 7.20 shows the frequency response of the input current to the EMI

filter, obtained with the RLC parameters reported in Fig. 7.17. The spectrum shows two resonance peaks and a maximum current draw of 4.64 mA.

The reduced value of this peak current does not lead to overheating of the filter components, since the associated dissipated power is reduced. Furthermore, such verification is pejorative for the following reasons:

1) The proposed system adopts a Digilent Analog Discovery 2 board which has a limitation on the maximum output current that can be supplied by the DAC channels at 4 mA.
2) In our simulation the measurement system is connected to the device under test only. In the real case, the generator is connected to a generic socket of the electrical system, therefore the current that can be supplied (4 mA) is distributed in the various parallel branches of the other connected devices, greatly reducing the intensity of the portion which could affect the EMI filters.

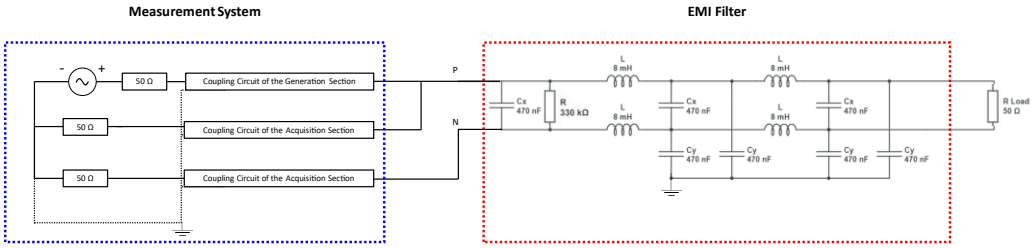

Figure 7.17: The scheme used for the SPICE simulation.

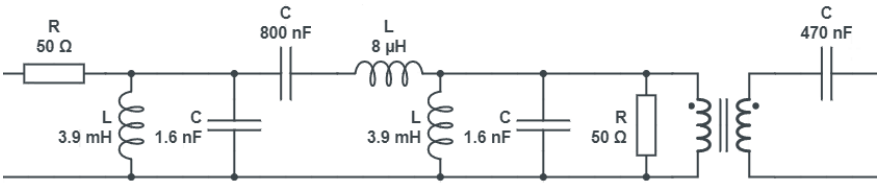

Figure 7.18: Coupling circuit for the signal generation section.

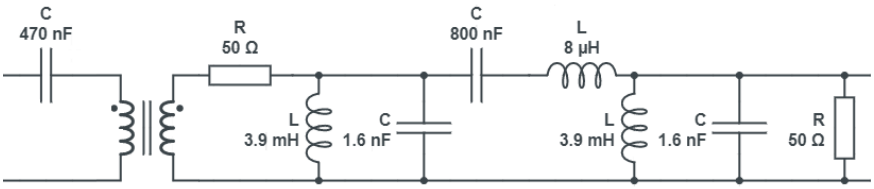

Figure 7.19: Coupling circuit for the signal acquisition section.

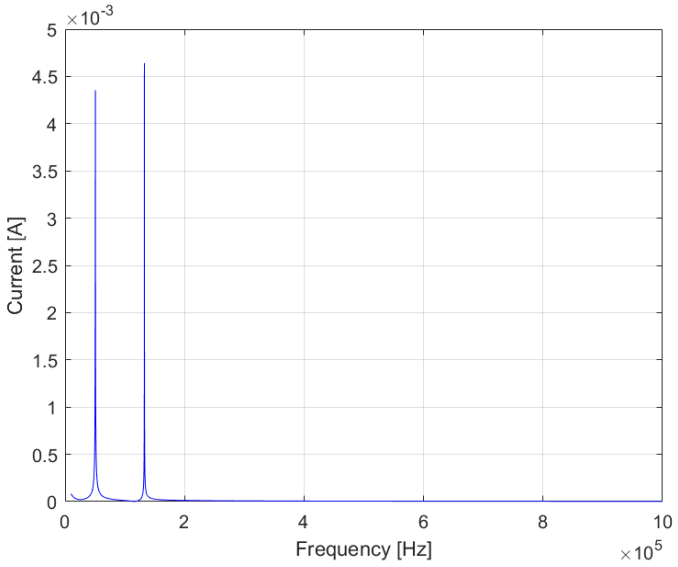

Figure 7.20: The frequency response of the input current to the EMI filter.

Chapter 8

Achievements and final remarks

This thesis project focuses on NILM systems, covering the design, implementation and metrological characterization of these systems. The study began by examining the state of the art of NILM systems. Due to its non-linear development process and the various directions it has taken, defining the current state of the art in the field of NILM systems research, which has garnered considerable interest from the scientific community, has been a challenging task.

The analysis of NILM systems was conducted by dividing them into event-based and non-event-based approaches. The former detect events by analyzing the electrical power signal or the current signal, while the latter process time windows of power or current signals and carry out energy disaggregations by recognizing certain consumption patterns. Chapter 2 provides a complete overview of the literature available on these systems, offering theoretical hints on the main models used, such as CO, DSC, FHMM and DL.

The thesis project identifies various applications in which NILM systems can be useful, including energy tariff recommendation systems, energy management systems for Smart Homes and Microgrids, Demand Response systems, anomaly detection and maintenance systems, energy disaggregation at the electrical substation level, and Ambient Assisted Living systems.

In order to conduct a comprehensive study, three systems were implemented, each with different characteristics and capabilities. All three proposed systems underwent an extensive testing phase, where their performance in real-world scenarios was evaluated. The systems differed in the type of approach, the different sampling frequencies used for signal acquisition, the characteristics of the hardware required, and the type of contribution they can make to the context in which they are applied.

Within NILM, performance analysis is typically conducted with reference to public datasets. This is an excellent comparative analysis, but unfortunately it does not allow for a correct metrological characterization of the system.

Unfortunately, no comparison with the (very few) advertised NILM systems was possible. The reasons are related to the fact that producers of energy disaggregation systems [15]-[17] sell an aggregate service and not the individual device.

For this reason, in addition to various NILM systems, a system that also allowed direct measurement of consumption of household appliances, based on a class 0.2 power meter, was also developed in order to obtain more reliable metrological data.

Final remarks on the NILM systems proposed in this thesis are given below:

1) The first proposed system, discussed in Chapter 5, is an event-based system that samples the absorbed current signal at a high frequency of 10 kHz. A typical event-based NILM system involves three main processes: signal acquisition (current and/or voltage), event detection, feature extraction, and load classification, in addition to signal acquisition (current and/or voltage). In contrast to other event-based NILM systems, which perform load classification based on the analysis of quantities also related to voltage (e.g., analysis in the P–Q or V–I plane [56]), the proposed system has the advantage of only measuring the overall current in a house. As a result, the complexity of the processing system is reduced. Another advantage is that the measuring system can be implemented as a galvanically isolated system at low cost using a clamp current transducer.

   In this work, the detection of an event and the classification of the related device were conducted by the same and unique process. The online system configuration (training) required approximately 7 min. The processing times measured were of the order of 105 ms for processing 1 s of acquired data (10K samples).

   The proposed NILM algorithm allows the system to recognize a device, regardless of whether it operates singularly or in combination with other loads. The first results obtained after a large number of measurements

appeared to be satisfactory, with error rates of approximately 3% for event detection and less than 2% for event classification.
The results obtained by processing the data available on the public BLUED data set appeared very encouraging. The value obtained for the F1-score was 99.8%, which is higher than that obtained with other systems using the same data set such as those proposed in [102] (91.5%) and [205] (93.2%).

2) The second system, discussed in Chapter 6, is a non-event-based system that processes an active power signal available at a significantly lower frequency of 1/8 Hz. The proposed system offers an embedded solution for energy disaggregation by leveraging a pre-trained DL model. Specifically, the model utilized is a sequence-to-point ANN, designed to predict the midpoint of the window of an appliance-level power reading from the corresponding aggregate power window.
The testing phase was conducted over 12 months, with the system installed in two houses in central Italy, one in Marche and one in Abruzzo, for six months each. During this time, the system processed the overall consumption of the houses to obtain details on individual consumption of dishwashers, washing machines, and fridge. The data analysis showed that the NILM system was able to adapt satisfactorily to both houses, even though the appliances had different absorption profiles. Additionally, the system could adapt when the same appliance had different duty cycles (as in the case of dishwasher and washing machine). The maximum relative percentage errors on a six-month basis found for dishwasher, washing machine, and fridge are 11%, 12%, and 10%, respectively.
Overall, the results of the study were highly satisfactory, especially when compared with prior research, where the model was evaluated offline on prerecorded data. The outcomes of this study were obtained by implementing the measurement system in real installations, where the model was applied to real-time acquired data. This confirmed the effectiveness and reliability of the model for on-field applications, including its generalization ability, applicability, and scalability. Such findings provide crucial insights into the practical implementation and adoption of the model.
The system was designed and implemented with the aim of demonstrating the feasibility of a solution based on a small-sized and low power microcontroller for real-time energy consumption monitoring. The microcontroller was chosen for its high performance and numerous

integrated peripherals, which allowed the development of a highly energy-efficient system. Thanks to its ARM Cortex-M7 architecture and a maximum clock frequency of 480 MHz, the microcontroller can handle complex DL algorithms while maintaining low power consumption.

The small size of the microcontroller, which measures only 20 mm x 20 mm, has allowed for the creation of a compact energy consumption monitoring system. This feature is particularly important in areas where a minimally invasive and low environmental impact solution is required, such as in the Internet of Things (IoT) where devices must be compact and non-intrusive.

It is important to note that the solution proposed in this thesis was not intended to develop a commercial product, but rather to demonstrate the feasibility of a solution based on a small-sized and low power microcontroller for real-time energy consumption monitoring. However, the small size and low power consumption of the proposed system make it potentially suitable for commercial energy consumption monitoring solutions.

As already stated in the introduction, access to commercial energy consumption monitoring solutions through NILM is still limited, especially for end consumers. Although there are companies developing NILM solutions [15]-[17], they tend to focus on business-to-business (B2B) services rather than direct business-to-consumer (B2C) hardware sales. This is mainly due to the fact that NILM technology is primarily used for energy management and monitoring in commercial and industrial settings, rather than in residential homes. These companies typically offer a wide range of services, such as energy audits, monitoring and reporting, and energy efficiency consulting, to businesses and organizations. This approach allows them to closely collaborate with customers to gain a deep understanding of their specific energy usage patterns and provide tailored solutions aimed at reducing energy consumption and costs. However, this presents a problem as it makes it difficult to compare the NILM systems proposed by research with commercially available NILM systems, as the latter are not readily accessible to the general public.

3) The third proposed system, discussed in Chapter 7, is particularly innovative and departs from the definitions of NILM systems used in the literature. This system is based on the injection of a variable-frequency signal into a generic electrical socket of the system being monitored. The SFRA

technique, already widely used in the diagnostics of transformers and asynchronous motors, has been applied in the third proposed system to characterize household appliances from the point of view of their influence in modifying the frequency response of the electrical system.

The obtained signature, influenced by physical characteristics of the loads, has been used as input for a machine learning algorithm, the SVM.

A large campaign of measurements was carried out on a test facility, during which eight different electrical loads were powered both individually and simultaneously. In particular, variable consumption loads such as a drill and a laptop were considered, which are generally among the most difficult for NILM systems to discriminate.

The proposed system demonstrated excellent performance even when trained with a minimum number of samples. The performances obtained are comparable with those obtained by other state-of-art electrical load identification systems [162],[206], in the face of a significant reduction in training data and a wider variety of loads. Furthermore, the proposed system allows to obtain information on which loads are powered in extremely short times (6.09 seconds in the final configuration of the system).

The advancement of NILM systems is an area of active research, as these systems need to improve their accuracy and reliability. In addition, there is a need to make these systems more accessible and user-friendly so that they can be easily adopted by homeowners, building managers, and utility companies.

The thesis project contributed to the advancement of the field of NILM systems, providing new insights into the practical implementation and adoption of NILM systems. By addressing key challenges related to data collection, preprocessing, and algorithm development, the project helped advance the state-of-the-art in NILM technology. As such, the work has the potential to contribute to the wider adoption of NILM systems.

List of Figures

List of Tables

References

[1] F. Ciancetta, G. Bucci, E. Fiorucci, S. Mari and A. Fioravanti, "A New Convolutional Neural Network-Based System for NILM Applications," in *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1-12, 2021, Art no. 1501112, doi: 10.1109/TIM.2020.3035193.

[2] G. Bucci, F. Ciancetta, E. Fiorucci, S. Mari and A. Fioravanti, "Measurements for non-intrusive load monitoring through machine learning approaches". *Acta IMEKO* 2021, 10, 90–96.

[3] S. Mari, G. Bucci, F. Ciancetta, E. Fiorucci, A. Fioravanti, "A Review of Non-Intrusive Load Monitoring Applications in Industrial and Residential Contexts". *Energies* 2022, 15, 9011. https://doi.org/10.3390/en15239011.

[4] S. Mari, G. Bucci, F. Ciancetta, E. Fiorucci, and A. Fioravanti, "A New NILM System Based on the SFRA Technique and Machine Learning," *Sensors*, vol. 23, no. 11, article 5226, 2023. https://doi.org/10.3390/s23115226.

[5] S. Mari, G. Bucci, F. Ciancetta, E. Fiorucci, A. Fioravanti, "An Embedded Deep Learning NILM System: A Year-long Field Study in Real Houses," in *IEEE Transactions on Instrumentation and Measurement* (**Under Review**).

[6] G. Bucci, F. Ciancetta, E. Fiorucci and S. Mari, "Load Identification System for Residential Applications Based on the NILM Technique," *2020 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*, 2020, pp. 1-6, doi: 10.1109/I2MTC43012.2020.9128599.

[7] A. Fioravanti, A. Prudenzi, G. Bucci, E. Fiorucci, F. Ciancetta and S. Mari, "Non intrusive electrical load identification through an online SFRA based approach," *2020 International Symposium on Power Electronics, Electrical Drives, Automation and Motion (SPEEDAM)*, 2020, pp. 694-698, doi: 10.1109/SPEEDAM48782.2020.9161856.

[8] G. Bucci, F. Ciancetta, E. Fiorucci, S. Mari, A. Fioravanti, "Deep Learning applied to SFRA Results: a Preliminary Study", *7th International Conference on Computing and Artificial Intelligence ICCAI 2021*, Tianjin, China, 23-26 April 2021, pp. 302-307 DOI: 10.1145/3467707.3467753.

[9] G. Bucci, F. Ciancetta, E. Fiorucci, S. Mari and A. Fioravanti, "Multi-State Appliances Identification through a NILM System Based on Convolutional Neural Network," *2021 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*, 2021, pp. 1-6, doi: 10.1109/I2MTC50364.2021.9460038.

[10] G. Bucci, F. Ciancetta, E. Fiorucci, S. Mari and A. Fioravanti, "A Non-Intrusive Load Identification System Based on Frequency Response Analysis," *2021 IEEE International Workshop on Metrology for Industry 4.0 & IoT (MetroInd4.0&IoT)*, 2021, pp. 254-258, doi: 10.1109/MetroInd4.0IoT51437.2021.9488472.

[11] G. Bucci, F. Ciancetta, E. Fiorucci, S. Mari, A. Fioravanti, "State of art overview of Non-Intrusive Load Monitoring applications in smart grids", *Measurement: Sensors* 18(2021) art. no. 100145. DOI: 10.1016/j.measen.2021.100145.

[12] S. Mari, G. Bucci, F. Ciancetta, E. Fiorucci and A. Fioravanti, "Advanced Architecture for Training and Testing NILM Systems," *2022 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*, 2022, pp. 1-6, doi: 10.1109/I2MTC48687.2022.9806577.

[13] M. Zeifman, C. Akers, and K. Roth, "Nonintrusive monitoring of miscellaneous and electronic loads," in Proc. IEEE Int. Conf. Consum. Electron. (ICCE), Jan. 2015, pp. 305–308.

[14] O. Parson, S. Ghosh, M. Weal, and A. Rogers, "Non-intrusive load monitoring using prior models of general appliance types," in Proc. Nat. Conf. Artif. Intell., Toronto, ON, Canada, Jul. 2012, pp. 1–8. [Online]. Available: http://eprints.soton.ac.uk/id/eprint/336812

[15] Voltaware. (2019). Intelligence for your home. Retrieved from https://www.voltaware.com

[16] Sense. (2018). Sense energy monitor. Retrieved from https://sense.com/product

[17] NET2GRID. (2021). NILM systems. Retrieved from website link https://www.net2grid.com/

[18] IEC TS 63297:2021 Sensing devices for non-intrusive load monitoring (NILM) systems.

[19] G. W. Hart. "Prototype nonintrusive appliance load monitor". In MIT Energy Laboratory Technical Report, and Electric Power Research Institute Technical Report. 1985.

[20] Zeifman, M.; Roth, K. Nonintrusive appliance load monitoring: Review and outlook. IEEE Trans. Consum. Electron. 2011, 57, pp.76–84.

[21] Y. Jin, E. Tebekaemi, M. Berges and L. Soibelman, "Robust adaptive event detection in non-intrusive load monitoring for energy aware smart facilities," *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2011, pp. 4340-4343, doi: 10.1109/ICASSP.2011.5947314.

[22] Z. Zheng, H. Chen, X. Luo, "A Supervised Event-Based Non-Intrusive Load Monitoring for Non-Linear Appliances". *Sustainability* 2018, *10*, 1001. https://doi.org/10.3390/su10041001.

[23] S. Dash, N.C. Sahoo, "Electric energy disaggregation via non-intrusive load monitoring: A state-of-the-art systematic review", *Electric Power Systems Research*, Volume 213, 2022, 108673, ISSN 0378-7796, https://doi.org/10.1016/j.epsr.2022.108673.

[24] M. Berges, E. Goldman, H. Matthews, L. Soibelman, "Learning systems for electric consumption of buildings", in *Proceedings of the 2009 ASCE International Workshop on Computing in Civil Engineering*, vol. 346, 2009, pp. 1–10, http://dx.doi.org/10.1061/41052(346)1.

[25] M.E. Berges, E. Goldman, H.S. Matthews, L. Soibelman, "Enhancing electricity audits in residential buildings with nonintrusive load monitoring", *Journal of Industrial Ecology* 14 (5) (2010) 844–858.

[26] M. Figueiredo, A. de Almeida, B. Ribeiro, "Home electrical signal disaggregation for non-intrusive load monitoring (NILM) systems", *Neurocomputing* 96 (2012) 66–73, Adaptive and Natural Computing Algorithms, https://doi.org/10.1016/j.neucom.2011.10.037.

[27] A. Prudenzi, "A neuron nets based procedure for identifying domestic appliances pattern-of-use from energy recordings at meter panel," *2002 IEEE Power Engineering Society Winter Meeting. Conference Proceedings (Cat. No.02CH37309)*, 2002, pp. 941-946 vol.2, doi: 10.1109/PESW.2002.985144.

[28] D. Srinivasan, W. S. Ng and A. C. Liew, "Neural-network-based signature recognition for harmonic source identification," in *IEEE Transactions on Power Delivery*, vol. 21, no. 1, pp. 398-405, Jan. 2006, doi: 10.1109/TPWRD.2005.852370.

[29] M. Baranski and J. Voss, "Genetic algorithm for pattern detection in NIALM systems," *2004 IEEE International Conference on Systems, Man and Cybernetics (IEEE Cat. No.04CH37583)*, 2004, pp. 3462-3468 vol.4, doi: 10.1109/ICSMC.2004.1400878.

[30] C.C. Yang, C.S. Soh, V.V. Yap, "A non-intrusive appliance load monitoring for efficient energy consumption based on naive Bayes classifier", *Sustainable Computing: Informatics and Systems* 14 (2017) 34–42, https://doi.org/10.1016/j.suscom.2017.03.001.

[31] C. Yang, C. Soh, V. Yap, "A systematic approach in appliance disaggregation using k-nearest neighbours and naive Bayes classifiers for energy efficiency", *Energy Efficiency* 11 (2018) http://dx.doi.org/10.1007/s12053-017-9561-0.

[32] Y.-H. Lin, Y.-C. Hu, "Electrical energy management based on a hybrid artificial neural network-particle swarm optimization-integrated two-stage non-intrusive load monitoring process in smart homes", *Processes* 6 (12) (2018) 236.

[33] A.S. Bouhouras, P.A. Gkaidatzis, E. Panagiotou, N. Poulakis, G.C. Christoforidis, “A NILM algorithm with enhanced disaggregation scheme under harmonic current vectors”, *Energy and Buildings* 183 (2019) 392–407.

[34] J. M. Gillis, S. M. Alshareef and W. G. Morsi, "Nonintrusive Load Monitoring Using Wavelet Design and Machine Learning," in *IEEE Transactions on Smart Grid*, vol. 7, no. 1, pp. 320-328, Jan. 2016, doi: 10.1109/TSG.2015.2428706.

[35] S.B. Leeb, S.R. Shaw, J.L. Kirtley, “Transient event detection in spectral envelope estimates for nonintrusive load monitoring”, *IEEE Transactions on Power Delivery* 10 (3).

[36] U.A. Khan, S.B. Leeb, M.C. Lee, “A multiprocessor for transient event detection”, *IEEE Transactions on Power Delivery* 12 (1) (1997) 51–60, http://dx.doi.org/10.1109/61.568225.

[37] S.R. Shaw, C.B. Abler, R.F. Lepard, D. Luo, S.B. Leeb, L.K. Norford, “Instrumentation for high performance nonintrusive electrical load monitoring”, *Transactions of the ASME, Journal of Solar Energy Engineering* 120 (3) (1998) 224–229, http://dx.doi.org/10.1115/1.2888073.

[38] C. Laughman, K. Lee, R. Cox, S. Shaw, S. Leeb, L. Norford, P. Armstrong, “Power signature analysis”, *IEEE Power and Energy Magazine* 1 (2) (2003) 56–63, http://dx.doi.org/10.1109/MPAE.2003.1192027.

[39] R. Cox, S. Leeb, S. Shaw, L. Norford, Transient event detection for nonintrusive load monitoring and demand side management using voltage distortion, in *Twenty-First Annual IEEE Applied Power Electronics Conference and Exposition*, 2006, APEC ’06, 2006, p. 7, http://dx.doi.org/10.1109/APEC.2006.1620777.

[40] S.R. Shaw, S.B. Leeb, L.K. Norford, R.W. Cox, “Nonintrusive load monitoring and diagnostics in power systems”, *IEEE Transactions on Instrumentation and Measurement* 57 (7) (2008) 1445–1454, http://dx.doi.org/10.1109/tim.2008.917179.

[41] S.R. Shaw, C.R. Laughman, “A Kalman-filter spectral envelope preprocessor”, *IEEE Transactions on Instrumentation and Measurement* 56 (5) (2007) 2010–2017, http://dx.doi.org/10.1109/TIM.2007.904475.

[42] H.-H. Chang, K.-L. Chen, Y.-P. Tsai, W.-J. Lee, “A new measurement method for power signatures of nonintrusive demand monitoring and load identification”, *IEEE Transactions on Industrial Applications* 48 (2) (2012) 764–771, http://dx.doi.org/10.1109/TIA.2011.2180497.

[43] M.-S. Tsai, Y.-H. Lin, “Modern development of an adaptive non-intrusive appliance load monitoring system in electricity energy conservation”, *Applied Energy* 96 (2012) 55–73, Smart Grids, https://doi.org/10.1016/j.apenergy.2011.11.027.

[44] Y.-H. Lin, M.-S. Tsai, "The integration of a genetic programming-based feature optimizer with fisher criterion and pattern recognition techniques to non-intrusive load monitoring for load identification", *International Journal of Green Energy* 12 (3)(2015) 279–290.

[45] S. Ghosh, A. Chatterjee, D. Chatterjee, "Improved non-intrusive identification technique of electrical appliances for a smart residential system", *IET Generation, Transmission & Distribution* 13 (5) (2018) 695–702.

[46] L.K. Norford, S.B. Leeb, "Non-intrusive electrical load monitoring in commercial buildings based on steady-state and transient load-detection algorithms", *Energy and Buildings* 24 (1) (1996) 51–64, http://dx.doi.org/10.1016/0378-7788(95)00958-2.

[47] L.-L. Lu-Lulu, S.-W. Park, B.-H. Wang, "Electric load signature analysis for home energy monitoring system", *The International Journal of Fuzzy Logic and Intelligent Systems* 12 (3) (2012) 193–197.

[48] Z. Wang, G. Zheng, "Residential appliances identification and monitoring by a nonintrusive method", *IEEE Transactions on Smart Grid* 3 (1) (2012) 80–92, http://dx.doi.org/10.1109/TSG.2011.2163950.

[49] L. Du, J.A. Restrepo, Y. Yang, R.G. Harley, T.G. Habetler, "Nonintrusive, self-organizing, and probabilistic classification and identification of plugged-in electric loads", IEEE Transactions on Smart Grid 4 (3) (2013) 1371–1380.

[50] Y.-C. Su, K.-L. Lian, H.-H. Chang, "Feature selection of non-intrusive load monitoring system using STFT and wavelet transform", in *2011 IEEE 8th International Conference on E-Business Engineering*, IEEE, 2011, pp. 293–298.

[51] S. Kong, Y. Kim, R. Ko, S.-K. Joo, "Home appliance load disaggregation using cepstrum-smoothing-based method", *IEEE Transactions on Consumer Electronics 61* (1) (2015) 24–30.

[52] Y.-H. Lin, M.-S. Tsai, "Development of an improved time-frequency analysis- based nonintrusive load monitor for load demand identification", *IEEE Transactions on Instrumentation and Measurement* 63 (6) (2013) 1470–1483.

[53] C.C. Yang, C.S. Soh, V.V. Yap, "A systematic approach in load disaggregation utilizing a multi-stage classification algorithm for consumer electrical appliances classification", *Frontiers in Energy* 13 (2) (2019) 386–398.

[54] J.D. Guedes, D.D. Ferreira, B.H. Barbosa," A non-intrusive approach to classify electrical appliances based on higher-order statistics and genetic algorithm: a smart grid perspective", *Electrical Power Systems Research* 140 (2016) 65–69.

[55] A.S. Kulkarni, C.K. Harnett, K.C. Welch, "EMF signature for appliance classification", *IEEE Sensors Journal* 15 (6) (2014) 3573–3581.

[56] T. Hassan, F. Javed, N. Arshad, "An empirical investigation of VI trajectory based load signatures for non-intrusive load monitoring", *IEEE Transactions on Smart Grid* 5 (2) (2013) 870–878.

[57] A.L. Wang, B.X. Chen, C.G. Wang, D. Hua, "Non-intrusive load monitoring algorithm based on features of V–I trajectory", *Electrical Power Systems Research* 157 (2018) 134–144.

[58] J. Froehlich, E. Larson, S. Gupta, G. Cohn, M. Reynolds, S. Patel, "Disaggregated end-use energy sensing for the smart grid", *IEEE Pervasive Computing* 10 (1) (2010) 28–39.

[59] J. Alcalá, J. Ureña, A. Hernández, D. Gualda, "Event-based energy disaggregation algorithm for activity monitoring from a single-point sensor", *IEEE Transactions on Instrumentation and Measurement* 66 (10) (2017) 2615–2626.

[60] H.-H. Chang, K.-L. Lian, Y.-C. Su, W.-J. Lee, "Power-spectrum-based wavelet transform for nonintrusive demand monitoring and load identification", *IEEE Transactions on Industrial Applications* 50 (3) (2014) 2081–2089, http://dx.doi.org/10.1109/TIA.2013.2283318.

[61] H.-H. Chang, L.-S. Lin, N. Chen, W.-J. Lee, "Particle swarm optimization based non-intrusive demand monitoring and load identification in smart meters", in *2012 IEEE Industry Applications Society Annual Meeting*, 2012, pp. 1–8, http://dx.doi.org/10.1109/IAS.2012.6373990.

[62] X. Wu, Y. Gao, D. Jiao, "Multi-label classification based on random forest algorithm for non-intrusive load monitoring system", *Processes* 7 (6) (2019) 337.

[63] Y. Liu, X. Wang, W. You, "Non-intrusive load monitoring by voltage–current trajectory enabled transfer learning", *IEEE Transactions on Smart Grid* 10 (5) (2018) 5609–5619.

[64] D. Yang, X. Gao, L. Kong, Y. Pang, B. Zhou, "An event-driven convolutional neural architecture for non-intrusive load monitoring of residential appliance", *IEEE Transactions on Consumer Electronics* 66 (2) (2020) 173–182.

[65] P. Held, S. Mauch, A. Saleh, D.O. Abdeslam, D. Benyoucef, "Frequency invariant transformation of periodic signals (fit-ps) for classification in nilm", *IEEE Transactions on Smart Grid* 10 (5) (2018) 5556–5563.

[66] Z. Fang, D. Zhao, C. Chen, Y. Li, Y. Tian, "Nonintrusive appliance identification with appliance-specific networks", *IEEE Transactions on Industrial Applications* 56 (4) (2020) 3443–3452.

[67] D. Zhou, Y. Wu, H. Zhou, "A nonintrusive load monitoring method for microgrid EMS using Bi-LSTM algorithm", *Complexity* 2021 (2021).

[68] L. Du, D. He, R.G. Harley, T.G. Habetler, "Electric load classification by binary voltage–current trajectory mapping", *IEEE Transactions on Smart Grid* 7 (1) (2015) 358–365.

[69] D. He, W. Lin, N. Liu, R.G. Harley, T.G. Habetler, "Incorporating non-intrusive load monitoring into building level demand response", *IEEE Transactions on Smart Grid* 4 (4) (2013) 1870–1877.

[70] L.R. Morais, A.R. Castro, "Competitive autoassociative neural networks for electrical appliance identification for non-intrusive load monitoring", *IEEE Access* 7 (2019) 111746–111755.

[71] S. Alshareef, W.G. Morsi, "Application of wavelet-based ensemble tree classifier for non-intrusive load monitoring", in *2015 IEEE Electrical Power and Energy Conference*, EPEC, IEEE, 2015, pp. 397–401.

[72] N. Sadeghianpourhamami, J. Ruyssinck, D. Deschrijver, T. Dhaene, C. Develder, "Comprehensive feature selection for appliance classification in NILM", *Energy and Buildings* 151 (2017) 98–106.

[73] Y.-H. Lin, M.-S. Tsai, "Non-intrusive load monitoring by novel neuro-fuzzy classification considering uncertainties", *IEEE Transactions on Smart Grid* 5 (5) (2014) 2376–2384.m

[74] P. Ducange, F. Marcelloni, M. Antonelli, "A novel approach based on finite-state machines with fuzzy transitions for nonintrusive home appliance monitoring", *IEEE Transactions on Industrial Informatics* 10 (2) (2014) 1185–1197.

[75] G. Elafoudi, L. Stankovic, V. Stankovic, "Power disaggregation of domestic smart meter readings using dynamic time warping", in *2014 6th International Symposium on Communications, Control and Signal Processing*, ISCCSP, IEEE, 2014, pp. 36–39.

[76] Y. Kim, T. Schmid, Z.M. Charbiwala, M.B. Srivastava, "Viridiscope: design and implementation of a fine grained power monitoring system for homes", in *Proceedings of the 11th International Conference on Ubiquitous Computing*, 2009, pp. 245–254.

[77] M. Dong, P.C.M. Meira, W. Xu, W. Freitas, "An event window based load monitoring technique for smart meters", *IEEE Transactions on Smart Grid* 3 (2) (2012) 787–796, http://dx.doi.org/10.1109/TSG.2012.2185522.

[78] B. Kalluri, A. Kamilaris, S. Kondepudi, H.W. Kua, K.W. Tham, "Applicability of using time series subsequences to study office plug load appliances", *Energy and Buildings* 127 (2016) 399–410.

[79] J.M. Gillis, W.G. Morsi, "Non-intrusive load monitoring using semi-supervised machine learning and wavelet design", *IEEE Transactions on Smart Grid* 8 (6) (2016) 2648–2655.

[80] J.M. Gillis, J.A. Chung, W.G. Morsi, “Designing new orthogonal high-order wavelets for nonintrusive load monitoring”, IEEE Transactions on Industrial Electronics 65 (3) (2017) 2578–2589.

[81] K. He, L. Stankovic, J. Liao, V. Stankovic, Non-intrusive load disaggregation using graph signal processing, *IEEE Transactions on Smart Grid* 9 (3) (2016) 1739–1747

[82] D. Li, S. Dick, “Residential household non-intrusive load monitoring via graph- based multi-label semi-supervised learning”, *IEEE Transactions on Smart Grid* 10 (4) (2018) 4615–4627.

[83] B. Zhao, K. He, L. Stankovic, V. Stankovic, “Improving event-based non-intrusive load monitoring using graph signal processing”, *IEEE Access* 6 (2018) 53944–53959.

[84] B. Qi, L. Liu, X. Wu, “Low-rate nonintrusive load disaggregation for resident load based on graph signal processing”, *IEEJ Transactions on Electrical and Electronic Engineering* 13 (12) (2018) 1833–1834.

[85] S. Welikala, N. Thelasingha, M. Akram, P.B. Ekanayake, R.I. Godaliyadda, J.B. Ekanayake, “Implementation of a robust real-time non-intrusive load monitoring solution”, *Applied Energy* 238 (2019) 1519–1529.

[86] G. W. Hart, "Nonintrusive appliance load monitoring," in *Proceedings of the IEEE,* vol. 80, no. 12, pp. 1870-1891, Dec. 1992, doi: 10.1109/5.192069.

[87] N. Batra, H. Dutta and A. Singh, "INDiC: Improved Non-intrusive Load Monitoring Using Load Division and Calibration," *2013 12th International Conference on Machine Learning and Applications*, 2013, pp. 79-84, doi: 10.1109/ICMLA.2013.21.

[88] J. Z. Kolter and M. J. Johnson, “REDD: A public data set for energy disaggregation research,” in *Workshop on Data Mining Applications in Sustainability* (SIGKDD), San Diego, CA, vol. 25, no. Citeseer. Citeseer, 2011, pp. 59–62.

[89] K. Ting, M. Lucente, G. S. Fung, W. Lee, and S. Hui, “A taxonomy of load signatures for single-phase electric appliances,” in *IEEE PESC (Power Electronics Specialist Conference)*, 2005.

[90] M. N. Schmidt, J. Larsen, and F. Hsiao. “Wind noise reduction using non-negative sparse coding”. In *IEEE Workshop on Machine Learning for Signal Processing*, 2007.

[91] M N. Schmidt and R. K. Olsson. “Single-channel speech separation using sparse non-negative matrix factorization”. In *International Conference on Spoken Language Processing*, 2006.

[92] P.O. Hoyer. “Non-negative sparse coding”. In *IEEE Workshop on Neural Networks for Signal Processing*, 2002.

[93] J. Eggert and E. Korner. “Sparse coding and NMF”. In *IEEE International Joint Conference on Neural Networks*, 2004.

[94] J. Z. Kolter, Siddharth Batra, and Andrew Y. Ng. “Energy Disaggregation via Discriminative Sparse Coding”. In *Advances in Neural Information Processing Systems*, 2010.

[95] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," in *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257-286, Feb. 1989, doi: 10.1109/5.18626.

[96] M. Zhong, N. Goddard, C. Sutton. “Signal aggregate constraints in additive factorial HMMs, with application to energy disaggregation”. In *Advances in neural information processing systems*. 2014, p. 3590–8.

[97] H. Kim, M. Marwah, M. Arlitt, G. Lyon, J. Han, “Unsupervised disaggregation of low frequency power measurements”, in *Proceedings of the 2011 SIAM International Conference on Data Mining*, SIAM, 2011, pp. 747–758.

[98] J.Z. Kolter, T. Jaakkola, “Approximate inference in additive factorial hmms with application to energy disaggregation”, in *Artificial Intelligence and Statistics*, PMLR, 2012, pp. 1472–1482.

[99] F. Paradiso, F. Paganelli, D. Giuli, S. Capobianco, “Context-based energy disaggregation in smart homes”, *Future Internet* 8 (1) (2016).

[100] R. Bonfigli, E. Principi, M. Fagiani, M. Severini, S. Squartini, F. Piazza, “Non-intrusive load monitoring by using active and reactive power in additive Factorial Hidden Markov Models”, *Applied Energy* 208 (2017) 1590–1607.

[101] S. Makonin, F. Popowich, I.V. Bajić, B. Gill, L. Bartram, “Exploiting HMM sparsity to perform online real-time nonintrusive load monitoring”, *IEEE Transactions on Smart Grid* 7 (6) (2015) 2575–2585.

[102] M. Aiad, P.H. Lee, “Unsupervised approach for load disaggregation with devices interactions”, *Energy in Buildings* 116 (2016) 96–103, https://doi.org/10.1016/j.enbuild.2015.12.043.

[103] Y. Li, Z. Peng, J. Huang, Z. Zhang, J.H. Son, “Energy disaggregation via hierarchical factorial hmm”, in *Proceedings of the 2nd International Workshop on Non-Intrusive Load Monitoring*, Austin, TX, USA, vol. 3, 2014, pp. 1–4.

[104] A. Géron, “Hands-on Machine Learning with Scikit-Learn, Keras and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems”. 2nd ed., O'Reilly, 2019.

[105] M. Sewak, S. K. Sahay, and H. Rathore, “Comparison of deep learning and the classical machine learning algorithm for the malware detection,” in Proc. 19th IEEE/ACIS Int. Conf. Softw. Eng., Artif. Intell., Netw.

Parallel/Distributed Comput. (SNPD), Busan, Jun. 2018, pp. 293–296, doi: 10.1109/SNPD.2018.8441123.

[106] Y. Zhang, Y. Huang, W. Li, and M. Lin, "Application of artificial neural network and DS algorithm to calibration transfer of rice protein powder," in Proc. 6th Int. Conf. Instrum. Meas., Comput., Commun. Control (IMCCC), Harbin, China, Jul. 2016, pp. 822–825, doi: 10.1109/IMCCC.2016.118.

[107] D. Jung, M. D. Nguyen, M. Park, J. Kim, and K. Mun, "Multiple classification of gait using time-frequency representations and deep convolutional neural networks," in IEEE Trans. Neural Syst. Rehabil. Eng., vol. 28, no. 4, pp. 997–1005, Feb. 2020, doi: 10.1109/TNSRE.2020.2977049.

[108] N.R. Draper and H. Smith, Applied Regression Analysis, 3rd ed. John Wiley, 1998. ISBN 0-471-17082-8.

[109] J. Brownlee. Loss and Loss Functions for Training Deep Learning Neural Networks0 Accessed: Jan. 28, 2019. [Online]. Available: https://machinelearningmastery.com/loss-and-loss-functions-fortraining-deep-learning-neural-networks/

[110] 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 (BuildSys '15), Association for Computing Machinery, New York, NY, USA, pp. 55–64, 2015. DOI: 10.1145/2821650.2821672

[111] M. Xia, W. Liu, K. Wang, X. Zhang, and Y. Xu, "Non-Intrusive Load Disaggregation Based on Deep Dilated Residual Network," Electr. Power Syst. Res., vol. 170, pp. 277–285, 2019.

[112] M. Kaselimi, E. Protopapadakis, A. Voulodimos, N. Doulamis, and A. Doulamis, "Multi-Channel Recurrent Convolutional Neural Networks for Energy Disaggregation," IEEE Access, vol. 7, pp. 81047–81056, 2019.

[113] C. Zhang, M. Zhong, Z. Wang, N. Goddard, and C. Sutton, "Sequence-to-point learning with neural networks for non-intrusive load monitoring," in Proceedings of the AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2018, pp. 2604–2611.

[114] S. Ahmed and M. Bons, "Edge Computed NILM: A Phone-Based Implementation Using MobileNet Compressed by Tensorflow Lite," in Proceedings of the 5th International Workshop on Non-Intrusive Load Monitoring (NILM'20), Association for Computing Machinery, New York, NY, USA, 2020, pp. 44–48.

[115] R. Kukunuri, A. Aglawe, J. Chauhan, K. Bhagtani, R. Patil, S. Walia, N. Batra, "EdgeNILM: Towards NILM on Edge Devices," in Proceedings of the 7th ACM International Conference on Systems for Energy-Efficient

Buildings, Cities, and Transportation (BuildSys '20), Association for Computing Machinery, New York, NY, USA, 2020, pp. 90–99.

[116] R. Bonfigli, A. Felicetti, E. Principi, M. Fagiani, S. Squartini, and F. Piazza, “Denoising Autoencoders for Non-Intrusive Load Monitoring: Improvements and Comparative Evaluation,” Energy Build., vol. 158, pp. 1461–1474, 2018.

[117] W. Kong, Z.Y. Dong, B. Wang, J. Zhao, and J. Huang, “A Practical Solution for Non-Intrusive Type II Load Monitoring Based on Deep Learning and Post-Processing,” IEEE Trans. Smart Grid, vol. 11, pp. 148–160, 2020.

[118] D. Murray, L. Stankovic, V. Stankovic, S. Lulic, and S. Sladojevic, "Transferability of Neural Network Approaches for Low-Rate Energy Disaggregation," in Proceedings of the ICASSP 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Brighton, UK, 12-17 May 2019, pp. 8330-8334.

[119] N. Miao, S. Zhao, Q. Shi, and R. Zhang, "Non-Intrusive Load Disaggregation Using Semi-Supervised Learning Method," in Proceedings of the 2019 International Conference on Security, Pattern Analysis, and Cybernetics (SPAC), Guangzhou, China, 20-23 December 2019, pp. 17-22.

[120] G. Cui, B. Liu, W. Luan, and Y. Yu, "Estimation of Target Appliance Electricity Consumption Using Background Filtering," IEEE Trans. Smart Grid, vol. 10, pp. 5920-5929, 2019.

[121] C. Shin, S. Rho, H. Lee, and W. Rhee, "Data Requirements for Applying Machine Learning to Energy Disaggregation," Energies, vol. 12, pp. 1696, 2019.

[122] M. D’Incecco, S. Squartini, and M. Zhong, "Transfer Learning for Non-Intrusive Load Monitoring," arXiv:1902.08835, 2019.

[123] C. Shin, S. Joo, J. Yim, H. Lee, T. Moon, and W. Rhee, "Subtask Gated Networks for Non-Intrusive Load Monitoring," arXiv:1811.06692, 2018.

[124] C. Brewitt and N. Goddard, "Non-Intrusive Load Monitoring with Fully Convolutional Networks," arXiv:1812.03915, 2018.

[125] J. Van Zaen, C.M. El Achkar, R.E. Carrillo, and A. Hutter, "Detection and Classification of Refrigeration Units in a Commercial Environment: Comparing Neural Networks to Unsupervised Clustering," in Proceedings of the 4th International Workshop on Non-Intrusive Load Monitoring, Austin, TX, USA, 7-8 March 2018, Nilmworkshop Org: Austin, TX, USA, 2018.

[126] O. Al Zeidi, "Deep Neural Networks for Non-Intrusive Load Monitoring," Master's Thesis, Monash University, Clayton, Australia, 2018.

[127] K. Chen, Q. Wang, Z. He, K. Chen, J. Hu, and J. He, "Convolutional Sequence to Sequence Non-Intrusive Load Monitoring," arXiv:1806.02078, 2018.

[128] 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, SETN '18, Patras, Greece, 9-12 July 2018, Association for Computing Machinery: New York, NY, USA, 2018, pp. 7:1-7:6.

[129] J. Jiang, Q. Kong, M. Plumbley, and N. Gilbert, "Deep Learning Based Energy Disaggregation and On/Off Detection of Household Appliances," arXiv, 2019, arXiv:1908.00941.

[130] A. Harell, S. Makonin, and I.V. Bajić, "Wavenilm: A Causal Neural Network for Power Disaggregation from the Complex Power Signal," arXiv, 2019, arXiv:1902.08736.

[131] P.B.M. Martins, J.G.R.C. Gomes, V.B. Nascimento, and A.R. de Freitas, "Application of a Deep Learning Generative Model to Load Disaggregation for Industrial Machinery Power Consumption Monitoring," in Proceedings of the 2018 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm), Aalborg, Denmark, 29–31 Oct. 2018, pp. 1-6, IEEE.

[132] A. van den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu, "WaveNet: A Generative Model for Raw Audio," arXiv, 2016, arXiv:1609.03499.

[133] Z. Jia, L. Yang, Z. Zhang, H. Liu, F. Kong, "Sequence to point learning based on bidirectional dilated residual network for non-intrusive load monitoring", International Journal of Electrical Power & Energy Systems, Volume 129, 2021, 106837, ISSN 0142-0615, https://doi.org/10.1016/j.ijepes.2021.106837.

[134] K. Chen, Y. Zhang, Q. Wang, J. Hu, H. Fan, and J. He, "Scale- and Context-Aware Convolutional Non-Intrusive Load Monitoring," IEEE Transactions on Power Systems, vol. 35, no. 3, pp. 2362-2373, 2020.

[135] 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, NILM'20, Yokohama, Japan, 18 Dec. 2020, pp. 89-93, Association for Computing Machinery.

[136] A.M. Sudoso and V. Piccialli, "Non-Intrusive Load Monitoring with an Attention-Based Deep Neural Network," arXiv, 2019, arXiv:1912.00759.

[137] M. Kaselimi, A. Voulodimos, E. Protopapadakis, N. Doulamis, and A. Doulamis, "EnerGAN: A Generative Adversarial Network for Energy Disaggregation," in Proceedings of the ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Barcelona, Spain, May 4-8, 2020, pp. 1578-1582.

[138] A. M. A. Ahmed, Y. Zhang, and F. Eliassen, "Generative Adversarial Networks and Transfer Learning for Non-Intrusive Load Monitoring in Smart Grids," in Proceedings of the 2020 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm), Tempe, AZ, USA, November 11-13, 2020, pp. 1-7.

[139] Y. Pan, K. Liu, Z. Shen, X. Cai, and Z. Jia, "Sequence-To-Subsequence Learning With Conditional Gan For Power Disaggregation," in Proceedings of the ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Barcelona, Spain, May 4-8, 2020, pp. 3202-3206.

[140] K. Bao, K. Ibrahimov, M. Wagner, and H. Schmeck, "Enhancing Neural Non-Intrusive Load Monitoring with Generative Adversarial Networks," Energy Inform., vol. 1, p. 18, 2018.

[141] 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.

[142] S. Darby, "The effectiveness of feedback on energy consumption," Orthod Fr., vol. 86, pp. 221-231, 2006. [Online]. Available: https://doi.org/10.1051/orthodfr/2015025.

[143] J. E. Fischer, S. Reece, E. Costanza, T. Rodden, N. R. Jennings, S. D. Ramchurn, M. Osborne, O. Parson, T. D. Huynh, M. Alam, N. Pantidi, S. Moran, and K. Bachour, ‘‘Recommending energy tariffs and load shifting based on smart household usage profiling,’’ in Proc. Int. Conf. Intell. User Interfaces, 2013, pp. 383–394.

[144] S. F. Baldwin et al., "Quadrennial Technology Review: An Assessment of Energy Technologies and Research Opportunities," U.S. Department of Energy (USA), September 2015.

[145] C. Joe-Wong, S. Sen, S. Ha, and M. Chiang, "Optimized Day-Ahead Pricing for Smart Grids with Device-Specific Scheduling Flexibility," IEEE J. Sel. Areas Commun., vol. 30, pp. 1075-1085, 2012. [Online]. Available: https://doi.org/10.1109/JSAC.2012.120706.

[146] SN Nambi, A.U. and R.V. Prasad, and A.R. Lua, "Decentralized Energy Demand Regulation in Smart Homes," IEEE Trans. Green Commun. Netw., vol. 1, pp. 372-380, 2017. [Online]. Available: https://doi.org/10.1109/TGCN.2017.2721818.

[147] 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 Trans. Smart Grid, vol. 12, pp. 977-987, 2021. [Online]. Available: https://doi.org/10.1109/TSG.2020.3027491.

[148] C. Xia, W. Li, X. Chang, F.C. Delicato, T. Yang, and A.Y. Zomaya, "Edge-based Energy Management for Smart Homes," in Proceedings of the 2018

IEEE 16th International Conference on Dependable, Autonomic and Secure Computing, 16th International Conference on Pervasive Intelligence and Computing, 4th International Conference on Big Data Intelligence and Computing and Cyber Science and Technology Congress (DASC/PiCom/DataCom/CyberSciTech), Athens, Greece, 12-15 August 2018, pp. 849-856. [Online]. Available: https://doi.org/10.1109/DASC/PiCom/DataCom/CyberSciTec.2018.00-19.

[149] D. Murray, L. Stankovic, and V. Stankovic, "An electrical load measurements dataset of United Kingdom households from a two-year longitudinal study," Sci. Data, vol. 4, article number 160122, 2017.

[150] Y. Liu, J. Ma, X. Xing, X. Liu and W. Wang, "A home energy management system incorporating data-driven uncertainty-aware user preference," Appl. Energy, vol. 326, pp. 119911, 2022. [Online]. Available: https://doi.org/10.1016/j.apenergy.2022.119911.

[151] R. Ramadan, Q. Huang, O. Bamisile, and A.S. Zalhaf, "Intelligent home energy management using Internet of Things platform based on NILM technique," Sustain. Energy Grids Netw., vol. 31, pp. 100785, 2022. [Online]. Available: https://doi.org/10.1016/j.segan.2022.100785.

[152] A. Lucas, L. Jansen, N. Andreadou, E. Kotsakis, and M. Masera, "Load flexibility forecast for DR using non-intrusive load monitoring in the residential sector," Energies, vol. 12, pp. 2725, 2019.

[153] W. Schneider and F. Campello de Souza, "Non-Intrusive Load Monitoring for Smart Grids," Technical Report, DELL EMC, Round Rock, TX, USA, 2018.

[154] M.R. Brambley, "A Novel, Low-Cost, Reduced-Sensor Approach for Providing Smart Remote Monitoring and Diagnostics for Packaged Air Conditioners and Heat Pumps," Technical Report, Pacific Northwest National Lab (PNNL), Richland, WA, USA, 2009.

[155] P.R. Armstrong, C.R. Laughman, S.B. Leeb and L.K. Norford, "Detection of rooftop cooling unit faults based on electrical measurements," HVAC&R Res., vol. 12, pp. 151-175, 2006.

[156] R.W. Cox, "Minimally Intrusive Strategies for Fault Detection and Energy Monitoring," Ph.D. Thesis, Massachusetts Institute of Technology, Cambridge, MA, USA, 2006.

[157] H. Rashid, P. Singh, V. Stankovic, and L. Stankovic, "Can non-intrusive load monitoring be used for identifying an appliance's anomalous behaviour?" Appl. Energy, vol. 238, pp. 796-805, 2019.

[158] P.J. Rousseeuw and L. Kaufman, "Finding Groups in Data," Wiley Online Library, Hoboken, NJ, USA, 1990.

[159] D. Green, T. Kane, S. Kidwell, P. Lindahl, J. Donnal, and S. Leeb, "NILM dashboard: Actionable feedback for condition-based maintenance," IEEE

Instrum. Meas. Mag., vol. 23, pp. 3-10, 2020. [Online]. Available: https://doi.org/10.1109/MIM.2020.9153467.

[160] A. Aboulian, D.H. Green, J.F. Switzer, T.J. Kane, G.V. Bredariol, P. Lindahl, J.S. Donnal, and S.B. Leeb, "NILM Dashboard: A Power System Monitor for Electromechanical Equipment Diagnostics," IEEE Trans. Ind. Inform., vol. 15, pp. 1405-1414, 2019. [Online]. Available: https://doi.org/10.1109/TII.2018.2843770.

[161] R. Beebe, "Estimate the increased power consumption caused by pump wear," Pump Mag., vol. 58, pp. 20-27, 2008.

[162] P.A. Lindahl, D.H. Green, G. Bredariol, A. Aboulian, J.S. Donnal, and S.B. Leeb, "Shipboard fault detection through non-intrusive load monitoring: A case study," IEEE Sens. J., vol. 18, pp. 8986-8995, 2018.

[163] D.H. Green, D.W. Quinn, S. Madden, P.A. Lindahl, and S.B. Leeb, "Nonintrusive Measurements for Detecting Progressive Equipment Faults," IEEE Trans. Instrum. Meas., vol. 71, pp. 3518112, 2022. [Online]. Available: https://doi.org/10.1109/TIM.2022.3193178.

[164] Y. Himeur, A. Alsalemi, F. Bensaali, and A. Amira, "Detection of Appliance-Level Abnormal Energy Consumption in Buildings Using Autoencoders and Micro-moments," in Proceedings of the International Conference on Big Data and Internet of Things, Morocco, North Africa, 17-18 March 2021, vol. 489. [Online]. Available: https://doi.org/10.1007/978-3-031-07969-6_14

[165] S. Dash and V. Venkatasubramanium, "Challenges in the industrial applications of fault diagnostic systems," Comput. Chem. Eng., vol. 24, pp. 785-791, 2000.

[166] S. Wang, R. Li, A. Evans, F. Li, "Regional nonintrusive load monitoring for low voltage substations and distributed energy resources," Appl. Energy, vol. 260, pp. 114225, 2020. [Online]. Available: https://doi.org/10.1016/j.apenergy.2019.114225.

[167] N. E. Huang et al. "The Empirical Mode Decomposition and the Hilbert Spectrum for Nonlinear and Non-Stationary Time Series Analysis," Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences, vol. 454, no. 1971, pp. 903-9195, 1998. [Online]. Available: http://www.jstor.org/stable/53161.

[168] S. Wang, R. Li, A. Evans, F. Li, "Electric Vehicle Load Disaggregation Based on Limited Activation Matching Pursuits," Energy Procedia, vol. 158, pp. 2611-2616, 2019. [Online]. Available: https://doi.org/10.1016/j.egypro.2019.02.011.

[169] A. Ruano, A. Hernandez, J. Ureña, M. Ruano, and J. Garcia, "NILM Techniques for Intelligent Home Energy Management and Ambient Assisted Living: A Review," Energies, vol. 12, pp. 2203, 2019.

[170] N. Noury, M. Berenguer, H. Teyssier, M. Bouzid, and M. Giordani, "Building an Index of Activity of Inhabitants from Their Activity on the Residential Electrical Power Line," IEEE Transactions on Information Technology in Biomedicine, vol. 15, no. 6, pp. 758-766, 2011. [Online]. Available: https://doi.org/10.1109/TITB.2011.2138149.

[171] X. Zhang, T. Kato, and T. Matsuyama, "Learning a context-aware personal model of appliance usage patterns in smart home," in Proceedings of the IEEE Innovative Smart Grid Technologies-Asia (ISGT ASIA), Kuala Lumpur, Malaysia, 2014, pp. 73-78. [Online]. Available: https://doi.org/10.1109/ISGT-Asia.2014.6873767.

[172] D.M. Blei, A.Y. Ng, and M.I. Jordan, "Latent dirichlet allocation," Journal of Machine Learning Research, vol. 3, pp. 993-1022, 2003.

[173] J. Alcalá, J. Ureña, Á. Hernández, and D. Gualda, "Assessing Human Activity in Elderly People Using Non-Intrusive Load Monitoring," Sensors, vol. 17, no. 3, pp. 351, 2017.

[174] J. Alcalá, J. Ureña, and Á. Hernández, "Activity supervision tool using Non-Intrusive Load Monitoring Systems," in Proceedings of the IEEE 20th Conference on Emerging Technologies & Factory Automation (ETFA), Luxembourg, 2015, pp. 1-4. [Online]. Available: https://doi.org/10.1109/ETFA.2015.7301622.

[175] P. Meier, K. Rohrmann, M. Sandner, M. Streitenberger, and M. Prochaska, “Convolutional neural networks for robust angular measurement with xMR sensor arrays,” in Proc. IEEE Int. Instrum. Meas. Technol. Conf. (I2MTC), May 2019, pp. 1–6, doi: 10.1109/I2MTC.2019.8827051.

[176] R. C. Gonzalez, “Deep convolutional neural networks [Lecture Notes],” IEEE Signal Process. Mag., vol. 35, no. 6, pp. 79–87, Nov. 2018, doi: 10.1109/MSP.2018.2842646.

[177] W. Yuegang, J. Shao, and X. Hongtao, “Non-stationary signals processing based on STFT,” in Proc. 8th Int. Conf. Electron. Meas. Instrum., Xi’an, China, Aug. 2007, pp. 3–301, doi:10.1109/ICEMI.2007.4350914.

[178] S. Zhang, D. Yu, and S. Sheng, “A discrete STFT processor for real-time spectrum analysis,” in Proc. IEEE Asia Pacific Conf. Circuits Syst., Singapore, Dec. 2006, pp. 1943–1946, doi: 10.1109/APCCAS. 2006.342241.

[179] D. Stursa and P. Dolezel, “Comparison of ReLU and linear saturated activation functions in neural network for universal approximation,” in Proc. 22nd Int. Conf. Process Control (PC19), Strbske Pleso, Slovakia, Jun. 2019, pp. 146–151, doi: 10.1109/PC.2019.8815057.

[180] S. Albawi, T. A. Mohammed, and S. Al-Zawi, “Understanding of a convolutional neural network,” in Proc. Int. Conf. Eng. Technol. (ICET),

Antalya, Turkey, Aug. 2017, pp. 1–6, doi: 10.1109/ICEngTechnol. 2017.8308186.

[181] K. Singh, A. Seth, H. S. Sandhu, and K. Samdani, "A comprehensive review of convolutional neural network based image enhancement techniques," in Proc. IEEE Int. Conf. Syst., Comput., Autom. Netw. (ICSCAN), Pondicherry, India, Mar. 2019, pp. 1–6, doi: 10.1109/ICSCAN.2019.8878706.

[182] O. Sheremet, K. Sheremet, O. Sadovoi, and Y. Sokhina, "Convolutional neural networks for image denoising in infocommunication systems," in Proc. Int. Sci.-Practical Conf. Problems Inf. Commun. Sci. Technol., Kharkiv, Ukraine, Oct. 2018, pp. 429–432, doi: 10.1109/INFOCOMMST. 2018.8632109.

[183] T. Katona and B. Antal, "Automated analysis of radiology images using convolutional neural networks," in Proc. 11th Int. Symp. Image Signal Process. Anal. (ISPA), Dubrovnik, Croatia, Sep. 2019, pp. 89–92, doi:10.1109/ISPA.2019.8868764.

[184] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," Proc. IEEE, vol. 86, no. 11, pp. 2278–2324, Dec. 1998.

[185] Anaconda Inc. (2020). Anaconda Software Distribution. Accessed: Feb. 13, 2020. [Online]. Available: https://www.anaconda.com/distribution/

[186] S. Makonin and F. Popowich, "Nonintrusive load monitoring (NILM) performance evaluation," Energy Efficiency, vol. 8, no. 4, pp. 809–814, Jul. 2015, doi: 10.1007/s12053-014-9306-2.

[187] J. L. Garcia-Balboa, M. V. Alba-Fernandez, F. J. Ariza-Lopez, and J. Rodriguez-Avi, "Homogeneity test for confusion matrices: A method and an example," in Proc. IEEE Int. Geosci. Remote Sens. Symp., Valencia, Spain, Jul. 2018, pp. 1203–1205, doi:10.1109/IGARSS.2018.8517924.

[188] K. Anderson, A. Ocneanu, D. Benitez, D. Carlson, A. Rowe, and M. Bergés, "BLUED: A fully labeled public dataset for event-based non-intrusive load monitoring research," in Proc. 2nd KDD Workshop Data Mining Appl. Sustainability (SustKDD), Beijing, China, Aug. 2012, pp. 1–5.

[189] C. Beckel, W. Kleiminger, R. Cicchetti, T. Staake, S. Santini, The ECO data set and the performance of non-intrusive load monitoring algorithms. Proceedings of the 1st ACM Conference on Embedded Systems for Energy-Efficient Buildings, 2014, pp. 80–89.

[190] C. Shin, E. Lee, J. Han, J. Yim, W. Rhee, H. Lee, The ENERTALK dataset, 15 Hz electricity consumption data from 22 houses in Korea, Sci. Data 6 (1) (2019) 1–13.

[191] 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.

[192] X. Ying, "An Overview of Overfitting and its Solutions," in Journal of Physics: Conference Series, vol. 1168, no. 2, p. 022022, 2019. DOI: 10.1088/1742-6596/1168/2/022022.

[193] M. Abadi et al., "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015. [Online]. Available: tensorflow.org. [Accessed on: Feb. 25, 2023].

[194] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," in Proceedings of the 3rd International Conference for Learning Representations, San Diego, 2015. DOI: 10.48550/arXiv.1412.6980.

[195] Node-RED, [Online]. Available: https://nodered.org/. [Accessed on: Feb. 25, 2023].

[196] Bureau International des Poids et Mesures, "JCGM 100: Evaluation of measurement data - Guide to the expression of uncertainty in measurement (GUM)," 1st ed., 2012.

[197] G. Bucci, F. Ciancetta and E. Fiorucci, "Apparatus for Online Continuous Diagnosis of Induction Motors Based on the SFRA Technique," in IEEE Transactions on Instrumentation and Measurement, vol. 69, no. 7, pp. 4134-4144, July 2020, doi: 10.1109/TIM.2019.2942172.

[198] IEC 60076-18:2012 Power transformers - Part 18: Measurement of frequency response

[199] "IEEE Guide for the Application and Interpretation of Frequency Response Analysis for Oil-Immersed Transformers" in IEEE Std C57.149-2012, vol., no., pp.1-72, 8 March 2013, doi: 10.1109/IEEESTD.2013.6475950.

[200] CEI EN 50160 – Power Quality Standard.

[201] F. D’Innocenzo, G. Bucci, S. Dolce, E. Fiorucci, F. Ciancetta “Power line communication, overview of standards and applications”. Proceedings of XXI IMEKO World Congress “Measurement in Research and Industry” August 30 - September 4, 2015, Prague, Czech Republic.

[202] T. Hofmann, B. Schölkopf, A. J. Smola, “Kernel methods in machine learning,” Ann. Statist., vol. 36, no. 3, pp. 1171–1220, 2008.

[203] S. Dolce, E. Fiorucci, G. Bucci, F. D’Innocenzo, F. Ciancetta, and A. Di Pasquale, “Test instrument for the automatic compliance check of cast resin insulated windings for power transformers,” Measurement, vol. 100, pp. 50–61, Mar. 2017.

[204] Schaffner Group, “Very High Performance Single-Phase Filters,” FN 2090 datasheet, 2017.

[205] A. K. Jain, S. S. Ahmed, P. Sundaramoorthy, R. Thiruvengadam, and V. Vijayaraghavan, "Current peak based device classification in NILM on a low-cost embedded platform using extra-trees," in *Proc. IEEE MIT Undergraduate Res. Technol. Conf. (URTC)*, Cambridge, MA, USA, Nov. 2017, pp. 1–4, doi: 10.1109/URTC.2017.8284200.

[206] B. Cannas, S. Carcangiu, D. Carta, A. Fanni, C. Muscas, G. Sias, B. Canetto, L. Fresi, P. Porcu, Real-Time Monitoring System of the Electricity Consumption in a Household Using NILM Techniques. In Proceedings of the 24th IMEKO TC4 International Symposium and 22nd International Workshop on ADC and DAC Modelling and Testing, Palermo, Virtual, Italy, 14–16 September 2020; pp. 90–95.