Intrusive and Non-Intrusive Load Monitoring (A Survey) Inference and Learning Approach

Danilo Burbano

Abstract—There is not discussion about the need of energy conservation, it is well known that energy resources are limited moreover the global energy demands will double by the end of 2030, which certainly will bring implications on the environment and hence to all of us.

Non-Intrusive load monitoring (NILM) is the process of recognize electrical devices and its energy consumption based on whole home electric signals, where this aggregated load data is acquired from a single point of measurement outside the household. The aim of this approach is to get optimal energy consumption and avoid energy wastage. Intrusive load monitoring (ILM) is the process of identify and locate single devices through the use of sensing systems to support control, monitor and intervention of such devices. The aim of this approach is to offer a base for the development of important applications for remote and automatic intervention of energy consumption inside buildings and homes as well.

Appliance discerns can be tackled using approaches from data mining and machine learning, finding out the techniques that fit the best this requirements, is a key factor for achieving feasible and suitable appliance load monitoring solutions. This paper presents common and interesting methods used.

Privacy concerns have been one of the bigger obstacles for implementing a widespread adoption of these solutions.

The implementation of security over these approaches along with fine-grained energy monitoring would lead to a better public agreement of these solutions and hence a faster adoption of such approaches.

Index Terms—**NILM; ILM; supervised learning; unsupervised learning; privacy; self-sensing; sensors; machine learning algorithms, smart meter; security**

Resumen—**La detección y predicción de la somnolencia es clave para la implementación de vehículos inteligentes destinados a prevenir accidentes en carreteras. Existen varios enfoques para crear este tipo de vehículos. En este artículo se analiza el enfoque de visión por computador, donde dispositivos embebidos son usados conjuntamente con técnicas de aprendizaje automático (i.e. inteligencia artificial) y reconocimiento de patrones para implementar soluciones para la detección del nivel de fatiga de un conductor de un vehículo. La mayora de investigaciones en este campo basados en visión por computador se enfocan en el análisis del parpadeo de los ojos del conductor, esta solución combinada con patrones adicionales como el reconocimiento del bostezo o el movimiento de cabeza constituye ser una solución bastante eficiente. El primer paso en este enfoque es el reconocimiento del rostro, para lo cual el uso del algoritmo AdaBoost muestra resultados precisos en el proceso de extracción de características, mientras para la detección de somnolencia el uso de clasificadores como el Support Vector Machine (SVM) muestra también resultados precisos.**

Un componente básico en la tecnología de visión por computador es el uso de una base de datos de imágenes espontaneas acorde al Facial Action Coding System (FACS), con la cual el clasificador puede ser entrenado. Este artículo presenta un prototipo sencillo para detección de somnolencia, en el cual el método de Viola-Jones es utilizado para el reconocimiento de rostros y un clasificador tipo cascada es usado para la detección de ojos cerrados en una secuencia continua de imágenes lo que constituye un indicador de somnolencia.

Palabras clave—**Somnolencia, Clasificador tipo cascada, método Viola-Jones, FACS, AU**

I. Introduction And Motivation

ONE of the most important innovations for reduction in the energy wastage is trough monitoring energy consumption. To achieve this, as in [2], research efforts have led to the development of Appliance Load Monitoring (ALM) aimed to perform detailed energy sensing and to provide information on the breakdown of the energy spent. Researches done so far, have split ALM in two approaches namely Intrusive Load Monitoring (ILM) and Non-Intrusive Load Monitoring (NILM), the former referred to a distributed sensing (i.e. inside the household) and the latter to a single point sensing (i.e. outside the household).

NILM examines the specific appliance signatures within the aggregated load data as was initially proposed by Hart [3], whereas the research for ILM approach has been addressed to ubiquitous sensor deployment due to its nature of distributed sensing, thereby allowing to identify NILM with smart meters devices and ILM with smart plugs.

It is important to stand out that there are some projects where the device meter is inside the house but is a single point of sensing at the same time, from an electrical point of view this can be classified as NILM, because are noise-based solution

This paper was submitted to the LAJC on 14th February 2015.

D. Burbano is with the Swiss Joint Master of Science in Computer Science Program, University of Bern, University of Neuchâtel an University of Fribourg, Switzerland (e-mail: daniluvatar@gmail.com)

that monitor the whole current operation of the house, but from a physical point of view this can fall inside ILM approach, for this paper this type of ALM will be classified depending of the propose of the solution, which means NILM for energy consumption and ILM for sensing systems.

NILM as well as ILM share some common characteristics, one of them is the use of machine learning techniques in order to predict the behavior of new appliance and to traduce the row data (e.g. current waveforms and voltage) in an easy and understandable form, these techniques also allow to made deeper analysis of the electricity consumption in such a way that it is even possible to know the behavior of the consumer, bringing privacy concerns into account.

This paper contribution is to show a general and arranged overview of these approaches, in order to make easy to understand for people not familiar with this terms, the general idea of this solutions, therefore helping those people to know exactly the specific subject to search for, in case of a deeper study is needed.

A main framework has been extracted from these projects for NILM and ILM as well, with focus on the machine learning techniques used for these approaches. This paper also shows the possibility to read the TV content through this techniques (obviously one of the most destabilizing issues that brings privacy concerns) in an attempt to avoid delay in the adoption of these good solutions, because knowing the weakness of this approaches is the starting point for finding a suited solution for this drawback.

The reminder of this paper is organized as follows. First Section 2 gives an overview of the related work about these approaches in energy sensing and inference and learning subjects. Section 3 gives an overview of the NILM and ILM frameworks. Section 4 and 5 introduces the most common machine learning algorithms used for electrical appliance configuration. Section 6 states the privacy concerns about these approaches given real scenarios of security.

II. Related Work

There are several works related to NILM and ILM, in which an assorted of device meters are used for the electrical signal analysis (i.e. energy sensing) and various solutions as well for appliance recognition (i.e. inference and learning).

A. Energy Sensing (Data Adquisition)

As was pointed out NILM is related to smart energy meters (low and high frequency) devices as it can discern devices from the aggregated data acquired from a single point of measurement [2], the aim of measure aggregated load have led to interesting projects like distributed NILM (dNILM) that uses an algorithm capable of running on limited meters with minimal bandwidth while still maintaining a high degree of accuracy [4], others works make use of different technologies like single electricity sensors as Ruzelli et al. that proposed the use of a single wireless energy monitoring sensor attached to the smart meter enabling real-time appliance recognition [6]. Another interesting work is an iterative way for separating individual appliances from an aggregated load as in [10].

On the other hand most of the work done so far about ILM is related to smart plugs, where technologies such as sensors and wireless are used aimed to get pervasive computing, Elzabadani et al. create a self-sensing space using Radio Frequency Identification (RFID) to recognize a plugged device and Open Services Gateway Initiative (OSGi) platform that facilitates the installation and operation of multiple services (i.e. appliances) on a single device [7]. Wook et al. were one step further and besides RFID, exploited the power draw characteristics of appliances using their unique current consumption as their IDs, along with a self-oriented platform known as Atlas [8]. A quite different work has been proposed by Zufferey et al. using plug-based low-end sensors which can automatically recognize home appliances based on their electric consumption profiles [9]. A very interesting work was made by Patel et al. that present an approach for a low-cost and easy to install power line that uses a single plug-in sensor to detect a variety of electrical events throughout the home [12].

An important research done that take advantage of new technologies as Gupta et al. that uses a noised-based solution through the electromagnetic influence (EMI) generated by switch mode power supplies (SMPS), an innovation found in modern consumer electronics and fluorescent lighting [5], this type of solution should fall inside NILM approach because is a single point of sensing, nevertheless it is also a device use in-home (connected to an outlet) therefore some could see it as ILM, although this can be seen as an hybrid work, in this paper the distinction between ILM an NILM will be made considering the sensing characteristic thus this work is consider in ILM approach.

It is important to stand out that works based in analyze the electrical noise on the power line like [5,12] born as an alternatively solution for ILM due to the high cost of one sensor per appliance, nevertheless this solutions also can be implemented for NILM, because it provides disaggregating electrical information for energy monitoring as claimed by its authors.

B. Inference and Learning (Classify Appliances)

There is an important work made by researches in order to find the best and feasible machine learning techniques used to identify appliance-specific states from the aggregated measurement. Before giving details about this works, it is important to stand out two main important works done over a prerequisite for any machine learning method, which is a database of load measurements, upon inference and learning methods can be applied. Kolter et al. [11] state key properties for such databases based in Reference Energy Disaggregation Data Set (REDD) [13].

According to the literature and works done about machine learning techniques with NILM systems, it is possible to classify in *supervised learning approach* and *unsupervised learning approach*, the former identify manually (i.e. labelling) the device change-points in a whole home energy signal or the system is trained on individual device power signals, whereas the latter consider the whole home signal without prior information (i.e. not labelling), and automatically separate signals [11].

Supervised learning works can be found in projects like Gupta et al. where events were automatically extracted and sent to a computer with labelling software afterwards inference was made using k-Nearest Neighbors algorithm (k-NN) using a 10-fold cross validation [5]. Ruzellie et al. proposed an approach that empowers users profiling where generated signatures are used to train an Artificial Neuronal Network (ANN) then employed to recognize appliance activities [6]. Kolter et al. monitor each individual circuit in the home focusing in Factorial Hidden Markov Model (FHMM) as a method for disaggregation with Expectation-Maximization (EM) for training [11]. Parson et al. use EM algorithm where data to train as well as appliance model are extracted from the aggregated load, the inference mechanism is done with an extension of the Viterbi algorithm able to iteratively disaggregate individual appliances [10].

Unsupervised learning have interesting works like dNILM approach uses event detection as a part of the learning phase where the meter detect abrupt changes in the power readings in order to classify appliances along with clustering algorithms and dynamic programming [4]. Gonçalves et al. used a genetic and K-means clustering algorithm for state changes and a matching pursuit algorithm to reconstruct the original power signals [15].

ILM approach has limited literature about the machine learning techniques used because in this approach there is often a middleware layer hiding this implementation, but as these solutions identifies each appliance through smart plugs [7,8], this process could be seen as labelling the signal therefore as supervised learning, also as in [9] where appliance recognition is based on supervised learning using two classification algorithms k-NN and Bayesian and in [12] that uses Support Vector Machine (SVM) performing classification by constructing an N-dimensional hyperplane that optimally separates the data into multiple categories. These works helps to infer that the middleware phase also can be seen as the encapsulation of even detection, feature extraction and learning.

III. NILM AND ILM FRAMEWORKS

Before given an explanation about the machine learning techniques used for disaggregation load as well as for learning or classifying the appliances in NLM and ILM approaches, a general overview of the architecture used in these approaches will be made in this section.

A. NILM Framework

The aim is to partition the whole house or building load energy data into its major constituents and estimate their power consumptions based on current voltage measurement of the main circuit entering a house. Figure 1 illustrates this approach summarized in phases or layers.

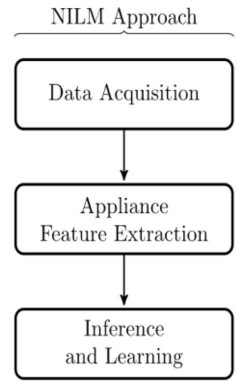

Figure 1. NILM framework

- DATA ACQUISITION phase acquires load measurement at an adequate rate so that distinctive load patterns can be identified. All smart meters nowadays at least have this feature.

- APPLIANCE FEATURE EXTRACTION phase processes the row data (i.e., voltage and current waveforms) in order to compute the power metrics (e.g., active and reactive power). Afterwards detect events such as appliance state transition (e.g., On to OFF) from the power measurements. An event detection module detects the ON/OFF transition of appliances by analyzing the changes in power levels. A proper goal of this phase would be to get an accurate appliance database or dataset as an input for the next phase.

- INFERENCE AND LEARNING phase analyzes the outputs of the previous phase. This phase can be divided in two approaches: *load identification* and *system training*, the former identify appliance-specific states from the aggregated measurement, the latter aim to training methods and learning of appliance features. As was pointed out in the Section 2, the main methods used for reach this aims are supervised and unsupervised algorithms.

B. ILM framework

Currently the aim of this approach is to pervasive computing (i.e. ambient intelligence, self-sensing spaces), where the main idea is to detect every single appliance with a smart plug, then middleware solutions are used to integrate and coordinate the whole nodes thus achieving a real time status and management of the household. Figure 2 illustrates this approach summarized in phases or layers.

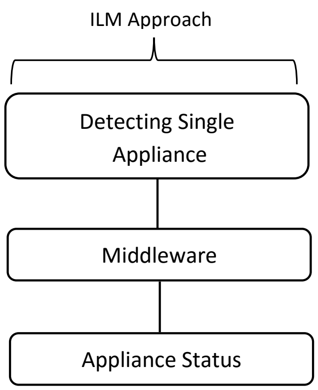

Figure 2. ILM framework

- DETECTING SINGLE APPLIANCE phase where using sensor technologies like smart plugs it is possible to identify the type as well as the location of the device that have been plugged. ILM projects have been used different technologies for this purpose like RFID and X10[1] protocols.
- MIDDLEWARE phase is related to the need of abstraction for every appliance by represent each physical device as a service in order to reach automatic identification and self-integration of appliances. The middleware solutions identified in works made so far are Atlas and OSGi platform. The personalized software used in projects as in [12] could also be considered in this phase.
- APPLIANCE STATUS phase is the final output of ILM, a user friendly interface that shows the real-time status of the household for control, intervention and monitoring.

A better understanding for achieving pervasive computing through ILM approach is illustrated in Figure 3, where there are three main components: i) Home Network, which is a wireless ad-hoc network of small devices. ii) Middleware, which is an orchestrator (i.e. coordinator system) of the nodes which compound the Home Network and provides the services (e.g. web services) to control such devices iii) Internet, the remote access that allows users to interact through an application (e.g. web pages, Android application) with their Home Network (i.e. smart house).

It is important to stand out that this approach originally does not consider devices connected directly to the live wire (i.e. non-plugged devices) such as light on a wall. This problem was stated in [8] although the detailed solution was not mentioned, but considering this, it is possible to infer that the need of measuring the current consumption showed up in such approaches. Thus analyze the signature of the current draw for identify the device with certainly addressed ILM to some machine learning techniques used in NILM and some solutions implemented in NILM that obtained the aggregated load from the live wire [6,11] can be projected to ILM as was already done in [9,5,12]. Works of machine learning techniques used in such works could be fall inside the Middleware phase.

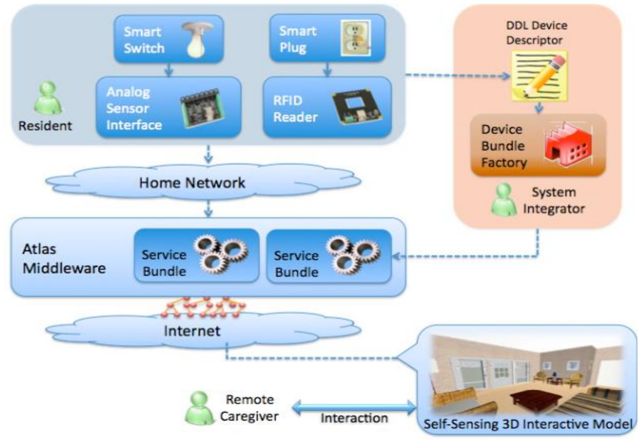

Figure 3. Pervasive computing trough ILM approach

IV. Database Establishment

The most important factor of enabling and advancing machine learning methods in any problem is the availability of proper databases.

A good reference is REDD, where the datasets can be categorized in high frequency (15kHz) for whole home data and low frequency (1 Hz) for individual circuit and plug data, the former can be used for NILM and the latter for ILM.

For supervised methods a good practice for establishment of a database would be to have tree dataset: (1) training set, (2) validation set and (3) test set.

New data has to be easily incorporated into the database for developing new methods and evaluating existing ones. On the other hand, the database should also enable the applications and innovations of various methods based on different features. Considering all these factors key properties such as informative, diverse and scalable should be defined in a database.

A. *Informative*

This property focuses on availability of information in the recordings of a database, such that a database is more informative if the data it contains can serve as training sets and test sets for more methods.

B. *Diverse*

The performance of a machine learning method is strongly influenced by within category differences and across category differences. A database has to be diverse enough to enable machine learning methods to capture these differences.

C. *Scalable*

[1] X10 is a protocol for communication among electronic devices used for home automation

New data has to be easily incorporated into the database. As the amount of data increases, the burden of using the database should be kept low, such that data processing, method modification, and comparison between methods before and after adding new data are easy.

A comprehensive database can be established gradually by merging data sets from different sources like single appliance data sets and whole main circuit data sets [13].

V. MACHINE LEARNING TECHNIQUES

As was pointed out in the Section 2 there are several machine learning techniques used for classification and estimation that can be broadly be divided in supervised and unsupervised learning. The former will be explained in subsection B and C, the latter in subsection D, E, F, G.

A. Problem Description

Formally, the aim of NILM is as follows. Given a discrete sequence of observed aggregate power readings x = $x_1,\ldots,x_T$, determine the sequence of appliance power demands $\mathbf{w}^{(n)} = w_1^{(n)},\ldots,w_T^{(n)}$, where n is one of N appliances.
Alternatively, this problem can be represented as the determination of appliances states, $\mathbf{z}^{(n)} = z^{(n)},\ldots,z_T^{(n)}$, if a mapping between states and power demands is known.
Each appliance state corresponds to an operation of approximately constant power draw (e.g. ‘on’, ‘off’ or ‘standby’) and *t* represents one of T discrete time measurements [10].

B. Training using Aggregate Data (Supervised Learning)

The most often model used is hidden Markov model (HMM) where data is represented as a sequence of observations, as can be seen in Figure 4.

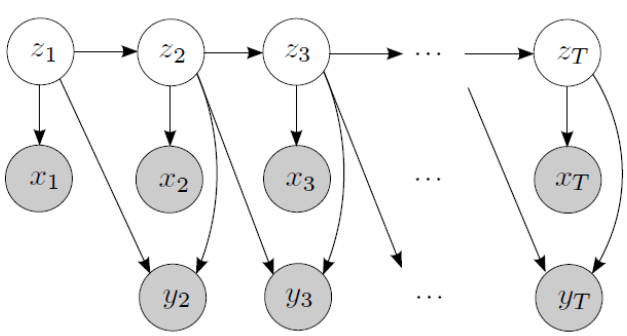

Figure 4. HMM variant. Shaded nodes represent observed variables and unshaded nodes represent hiden variables

z_T represents the state of appliance (e.g. on/off) at an instant time whereas *x* is an observation sequence (e.g. vector of electrical parameters such as active and reactive power) corresponding to the house aggregate power demand measured by the smart meter and *y* corresponds to the difference between two consecutive aggregate power readings such that $y_t = x_t - x_{t-1}$. These derived observations are used to infer the probability that a change in aggregate power, y_t, was generated by two consecutive appliance states [10]. It is important to stand out that the standard HMM model does not consider the y observation.

This model has been translated from a simpler model of state transitions like the one show in Figure 5a.

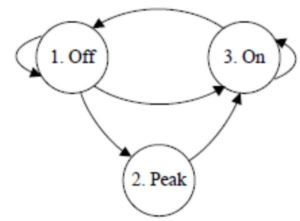

Figure 5a. State transition model

Supervised training methods use prior knowledge of appliance behavior and power demands, the generic model of an appliance type consists of prior over each parameter (i.e. active and reactive power) of an appliance’s model. This prior is simply an expected normal operation of a device, as represented in Figure 5b.

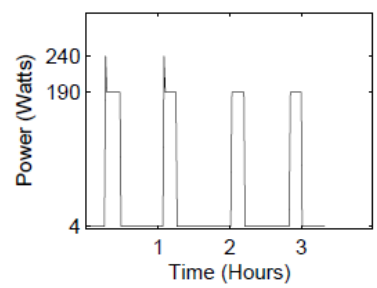

Figure 5b. Prior of appliance power

An appliance prior should be general enough to be able to represent many appliance instances of the same type (e.g. all refrigerators) [10]. From this information (Figure 5a and Figure 5b), values for the appliance’s prior can be inferred as shown in Figure 5c.

z_t	μ	σ^2
1	4 W	10
2	240 W	150
3	190 W	100

Figure 5c. Emission density

These prior models could be determined in a number of ways. Most directly, a domain expert could estimate an appliance’s emission density using knowledge of its power demand available from appliance documentation. Alternatively, these parameters may be constructed by generalizing data collected from laboratory trials or other sub-metered homes [10].

As supervised method requires a clean dataset for training the classifier, in order to achieve this many projects focus on exploit periods during which a single appliance turns on and off without any other appliances changing state, because this allow to have a signature in the aggregate load which is unaffected by all other appliances apart from the baseline load.

Figure 6 shows an example of the aggregate power demand. From hours 1 to 3 it is clear that only the refrigerator is cycling on and off. This period can be used to train the refrigerator appliance model, which is then used to disaggregate the refrigerator's load for the whole duration [10].

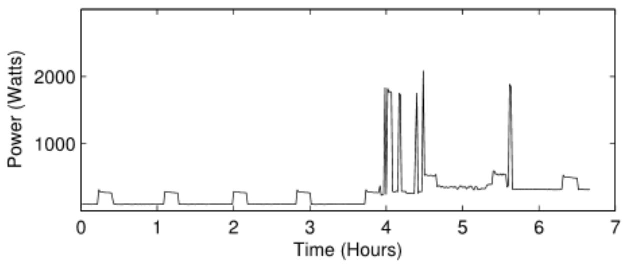

Figure 6. Example of aggregated power demand

C. *Energy Disaggregation (Supervised Learning)*

Subtracting the refrigerator's load will consequently clean the aggregate load allowing additional appliances to be identified and disaggregated. This subtraction can be achieved by running the EM algorithm on small overlapping windows of aggregate data.

The disaggregation task aims to infer each appliance's load given only the aggregate load and the learned appliance's parameters (Figure 5c). After learning the parameters for each appliance which we wish to disaggregate, any inference mechanism capable of disaggregating a subset of appliances could be used, for instance k-NN, EM, Viterbi algorithms. In general these algorithms can be found in machine learning literature, thus research about machine learning techniques that best fit the requirements of this type of aggregated load data has been the focus of research about NILM.

D. *Event Detection Algorithm (Unsupervised Learning)*

As a part of both the learning phase and the monitoring phase, the first goal of the meter is to detect abrupt changes in the power readings which correspond to loads changing state. Thus, the event detector produces an output whenever there is sufficient change from the current power value and by ignoring minor changes, meter readings collected every second can be represented in a compressed form. The input is a series of power data tuples like {time stamp, real power value}.

The algorithm uses a series of tuples as the input and creates a series of tuples with the same format for the output. These output tuples represent sudden changes in power, produced when an appliance changes state. Figure 7 illustrates the output produced by running the event detector algorithm against lab data.

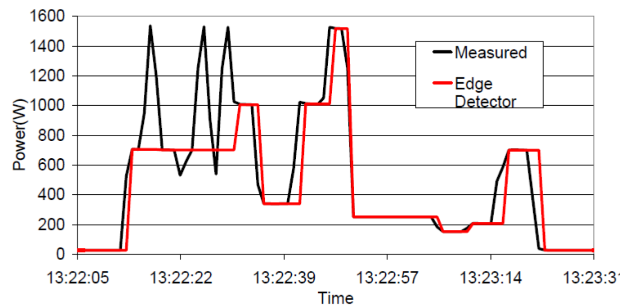

Figure 7. Event Detection Algorithm

E. *The Learning Phase (Unsupervised Learning)*

Power changes must be identified in order to classify what large appliances are present and group each power change with the respective load. Once the loads are identified, they are placed in a static table and returned to the meters to be used during the real-time phase.

Considering that the most basic load is on/off model, the first step to analyze the data on the backend controller is to establish clusters of on and off events in order to begin identifying what appliances are being seen by the meters.

F. *Clustering Algorithm (Unsupervised Learning)*

This algorithm accomplishes the establishment of clusters of on and off events by taking an interval of data and grouping like events by their respective power changes. The algorithm retrieves the first element of the event list, determines the power change, and then searches the rest of the array for matching events by assigning the first event to a new cluster. The algorithm also returns how many times each cluster event was turned on or off during the collection period.

It is important to stand out that in real world scenarios, not all loads can be explained by just on/off states, therefore some solutions for accurately identify present loads is building finite state machines (FSM) based upon genetic algorithms with dynamic programming, which means that the table returned to the meters should be a dynamic one.

G. *When to Learn (Unsupervised Learning)*

The schedule for learning consists of both reactive learning initiated from the meters and proactive learning initiated from the backend controller. When either detects the need for learning it informs the other and the meters begin to transmit data until the controller indicates that it has sufficient information to build a static table. Figure 8a shows an example of a static table rebuilt after a new appliance has been detected as is illustrated in Figure 8b.

Appliance	#States	State 1 Real Power (W)		State 2 Real Power (W)	
		Mean	Std	Mean	Std
(A) Dryer	2	5594	200	2510	90
(B) Oven 1	1	5733	65	NA	NA
(C) Oven 2	1	2215	74	NA	NA
(D) Garage	1	961	55	NA	NA
(E) Toaster	1	1509	45	NA	NA

Relearning

Figure 8a. Static table

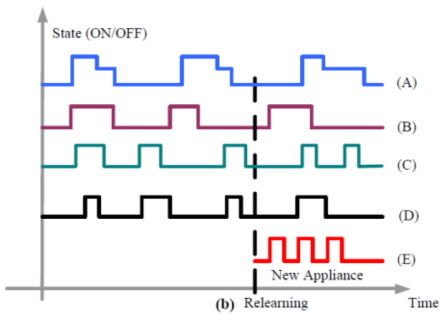

Figure 8b. Appliance States

H. Comparison of load disaggregation algorithms

As was stated in different sections of this paper, there are several algorithms used for the task of disaggregation from the aggregated load, this techniques can be applied to NILM and ILM approach as well, in order to make easy further research of this approaches a summarized and categorized list of this algorithms is done in Table I [2].

TABLE I. Disaggregation Algorithms

Learning Algorithm	Accuracy %	Training S[a]/U[b]	Real-Time
SVM	75-98	S	Yes
Bayes	80-99	S	Yes
HMM	75-95	B	No
Neural Networks	80-97	S	Yes
k-NN	70-90	S	Yes

a. Supervised. (S)

b. Unsupervised. (U)

c. Both (B)

VI. Privacy Concerns

Concerns about NILM and ILM approaches showed up since smart meters are able to become surveillance devices that monitor the behavior of the customers, which could lead to unprecedented invasions of consumer privacy as in [14] where issues about security and privacy are shown.

A. Data Transmission

An experiment was done with a real provider of smart meters, Discovergy[2]. All data are transmitted (Figure 9a) to the servers hosted by Discovergy, and then customers are able to access these data via a web-browser.

POST /api/w.html HTTP/1.1
Content-Type: application/x-www-form-urlencoded
Host:85.214.93.99
Content-Length:851
version=0.9&identity= [redacted] &msg=228601&values=[
{"meterdata":"00000285.9823514*kWh","tickdelta":"00000285.9822239*kWh","seconds":"399511319.61"},
{"meterdata":"00000285.9824793*kWh","tickdelta":"00000285.9823514*kWh","seconds":"399511321.61"},
{"meterdata":"00000285.9826075*kWh","tickdelta":"00000285.9824793*kWh","seconds":"399511323.61"},
{"meterdata":"00000285.9827358*kWh","tickdelta":"00000285.9826075*kWh","seconds":"399511325.62"},
{"meterdata":"00000285.9828636*kWh","tickdelta":"00000285.9827358*kWh","seconds":"399511327.62"},
{"meterdata":"00000285.9829915*kWh","tickdelta":"00000285.9828636*kWh","seconds":"399511329.62"},
{"meterdata":"00000285.9831196*kWh","tickdelta":"00000285.9829915*kWh","seconds":"399511331.62"},
{"meterdata":"00000285.9832476*kWh","tickdelta":"00000285.9831196*kWh","seconds":"399511333.62"}]
&now=399511335.65

Figure 9a. Captured communication between smart meter and server

The energy consumption data are saved in a text file format, while being transferred to the central servers. The unencrypted data can be easily hacked out; in addition none of the data are signed. The identity (highlighted in black in Figure 9a) of any smart meter is immediately revealed when the data are being transmitted to the central servers and could be used by an attacker to send different power consumption data to the server.

A java-script based application requests the data from the Discovergy server and offers the visualization of the profile. A typical profile example can be seen in Figure 9b.

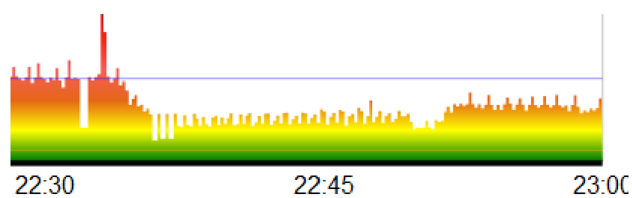

Figure 9a. Captured communication between smart meter and server

B. TV/Film Detection

The test was made with an LCD television which uses the display technology to produce colored images. The core of the content identification program detailed in [14] is the power consumption prediction function. The input of the function is the multimedia content; the output is power usage prediction as would be displayed by a smart meter.

The first step is to measure the power consumption for a series of pictures consisting of elementary shades. The additive RGB color notation with one byte (i. e. values 0–255) is used that increase the brightness from black to white running over 254 shades of gray. The experiment shows that maximum power consumption is reached with rather dark pictures (e. g., RGB 32-32-32) but this also depends on the television user settings. The next step is to extract frames from the movie and calculate the brightness of each frame.

As the experimental results were obtained with a smart meter operating on a two-seconds interval, calculations were made with an average value of power consumption for a number of consecutive frames adding up to two seconds of a movie, which means 50 frames for a movie with a typical 25 frames per second (fps) rate. Figure 4 shows the final result.

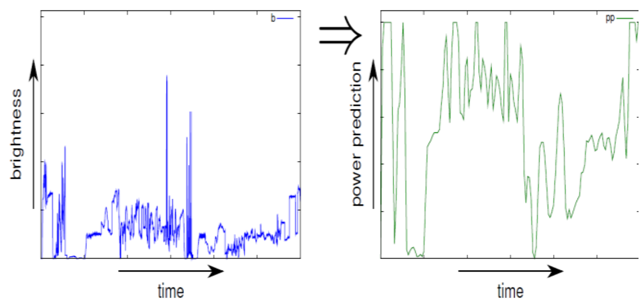

Figure 10. Power prediction is computed on frame brightness values

Experiments done with 653 content files allowed to state that detection of movies produced for cinema projectors was almost always a feasible task while many TV studio productions (e. g. talk shows, news) are difficult or impossible to identify when played as recorded content. Though it is still rather easy to determine which TV broadcast consumers are watching because just five minutes of consecutive playing of a movie is in many cases sufficient to identify the viewed content by analyzing the smart meter power consumption data along with the content of live stations besides Greveler et al. showed up that the content identification is also possible with CR tube and plasma TVs [14].

C. Security over NILM

As was shown in the subsection above, the need of security over this approach is really necessary, this is acknowledged for the scientific community, therefore there are currently two approaches of implementing privacy preserving smart meter data analysis.

The first approach relies on masking the meter readings, and the second approach relies on homomorphic encryption detailed information can be seen in [14].

VII. CONCLUSION

The advancements in the works of NILM and ILM are increasing diary as any other technology, it certainly will lead us to a broadly adoption of smart meters and self-sensing spaces once fine-grained energy monitoring is achieved, although there are public concerns about the use of this information considering the possibility of a misuse of this data by thirds parties chasing the electricity consumption of a customer is latent, because it can be used for marketing attack and even worst government or malicious eavesdropping. This opens the door to an inevitable and needed research of security over NILM and ILM.

Proper datasets is a prerequisite for enabling and advancing machine learning methods, therefore availability of those databases is transcendental to reach feasible machine learning techniques.

There are several machine learning methods used in for NILM and ILM for recognizing appliances, the most promised result are in the Neural Networks and Bayes approaches, although the works in variants of HMM, clustering and genetic algorithms focus on unsupervised learning are very important as it does not imply a training phase.

According to the information that has been pointed out in this paper, it is possible to infer that researches used some machine learning techniques to predict the power consumption from the brightness of a TV and in the same way it would be possible to made an inverse predicted function (i.e. from the power predict the brightness) thus allowing to identify the TV watching habits from a person with NILM or ILM approaches, another way to get this information from the customer would be having a database of the power metric movies (predicted from its brightness) and match with the power metrics done by the smart meter.

Making a projection of what can be accomplished for, after a fine-grained monitoring and control of appliance using ILM is a Smart House to provide healthcare, convenience, entertainment, energy efficiency and security being gentile with the environment at the same time. The same projection can be made by NILM approach that is and will be a key component of Smart Grid to improve the efficiency and sustainability of the production and distribution of electricity avoiding the wastage of electricity and thus improving the consumer economy and even most important helping to have a more health environment.

According to the information presented in this paper and the conclusions that have been made, it would be very important that most of the future work and research about NILM and ILM solutions focus on mainly two approaches:

1) *Security:* Methods of security over these approaches in order to find feasible solutions like encrypted data, which allow offering a reliable service and therefore the people approval for these solutions.

2) *Unsupervised learning*: The use of this machine learning technique, certainly will lead to a faster adoption of these approaches because as it does not need setup of instruments and human intervention for learning, it allows an efficient use of resources and hence a feasible solution from an economic point of view.

ACKNOWLEDGMENT

Thanks to all the scientific community that had been involve in the works and projects about NILM and ILM, all those researches indirectly helped to accomplish this work.

REFERENCES

[1] Uteley, J.; Shorrock, L. Domestic Energy Fact File 2008; Technical Report for Building Research Establishment: Garston, UK, 2008.

[2] Ahmed Zoha, Alexander Gluhak, Muhammad Ali Imra & Sutharshan Rajasegarar; Non-Intrusive Load Monitoring Approaches for Disaggregated Energy Sensing:A Survey, 2012.

[3] Hart, G.W. Nonintrusive appliance load monitoring. IEEE Proc. 1992, 80, 1870–1891.

[2] https://discovergy.com

[4] David C. Bergman, Dong Jin, Joshua P. Juen, Naoki Tanaka, Carl A. Gunter, Andrew K. Wright; Distributed Non-Intrusive Load Monitoring, 2011.

[5] Sidhant Gupta, Matthew S. Reynolds, Shwetak N. Patel; ElectriSense: Single-Point Sensing Using EMI for Electrical Event Detection and Classification in the Home, 2010.

[6] A.G. Ruzzelli, C. Nicolas†, A. Schoofs, and G.M.P. O'Hare; Real-Time Recognition and Profiling of Appliances through a Single Electricity Sensor.

[7] Hicham Elzabadani, Abdelsalam (Sumi) Helal, Bessam Abdulrazak and Erwin Jansen; Self-Sensing Spaces: Smart Plugs For Smart Environments, 2009.

[8] Jung Wook Park, Chao Chen, Hicham Elzabadani, Sumi Helal; SmartPlug: Creating Self-Sensing Spaces using Atlas Middleware, 2009.

[9] Damien Zufferey , Christophe Gisler, Omar Abou Khaled, JeanHennebert; Machine Learning Approaches For Electric Appliance Classification, 2012.

[10] Oliver Parson, Siddhartha Ghosh, Mark Weal, Alex Rogers; Non-intrusive Load Monitoring using Prior Models of General Appliance Types, 2012.

[11] J. Zico Kolter, Matthew J. Johnson; REDD; A Public Data Set for Energy Disaggregation Research, 2011.

[12] Shwetak N. Patel, Thomas Robertson, Julie A. Kientz, Matthew S. Reynolds, and Gregory D. Abowd; At the Flick of a Switch: Detecting and Classifying Unique Electrical Events on the Residential, 2007.

[13] Po-Hsiang Lai, Mark Trayer, Sudhir Ramakrishna, and Ying Li; Database Establishment for Machine Learning in NILM Power Line.

[14] Ulrich Greveler, Peter Glosekotterz, Benjamin Justusy, Dennis Loehry; Multimedia Content Identification Through Smart Meter Power Usage Profiles, 2011.

[15] Hugo Gonçalves, Adrian Ocneanu, Mario Bergés; Unsupervised disaggregation of appliances using aggregated consumption data, 2011.

Danilo Burbano Acuña received his B.S. degree in systems engineering from National Polytechnic School, Ecuador in 2006. From 2006 to 2013 he was a system analyst and software developer in three different enterprises. Starting in a banking software development company, then in a massive beverage production company and then in a government company which controls and administrates the taxes around Ecuador. He is currently pursuing the M.S. degree in computer science with a specialization in distributed systems in a Swiss joint master program at the University of Bern, Neuchâtel and Fribourg. He has been coauthor of papers regarding the internet of things and is currently working in his master thesis about indoor tracking. His research interest