Non-Intrusive Load Monitoring Using Artificial Intelligence Classifiers: Performance Analysis of Machine Learning Techniques

R. V. A. Monteiro, J. C. R. de Santana, R. F. S. Teixeira, A. S. Bretas, R. Aguiar, C. E. P. Poma

***Abstract*--In recent years, strategies for load monitoring have been proposed to mitigate power consumption. It has been found, in several reported studies, that as more information is provided for consumers about their electricity consumption, more power energy conservation will occur. In this way, Non-Intrusive Load Monitoring (NILM) has been studied and applied in real-life applications. It consists of detecting and classifying appliances on/off states by measuring electrical signals only at one location of the residential consumer. Several studies have been made using different techniques to improve the accuracy of this strategy. In this paper electromagnetic transients are taking into account and, a performance analysis between cutting-edge artificial classifiers is made. It has been found that 1D convolutional neural networks perform better for this case and electrical current signals are more suitable for NILM, once it carries more features than voltage and power signals.**

***Keywords*: NILM; Electromagnetic Transients; Deep learning; Artificial Intelligence; Energy management.**

I. Introduction

The conservation of electric energy, as it currently stands, is a technical challenge for power systems engineers, which have proposed new energy conservation programs, technologies or methods towards consumption of electric energy minimization. Within this study, one can highlight the Brazilian residential sector, which in 2017 was responsible for 28.86% of the total electric energy consumed in the country [1]. This is a growing concern, since energy resources are limited and the increase in consumption within the electric energy sector has a direct impact, that is normally negative, on the natural environment, as for example emissions of CO_2 [2].

A significant reduction in the consumption of electric energy can be obtained through detailed information sent to consumers of their own energy use. This can be seen through collected and published material that took into consideration more than 60 studies [3]. [3] makes a case that the maximum consumer economy of electric energy can be reached if details of real-time consumption are made available, different to traditional monthly billing of electric energy [3].

The existing techniques that allow for this real-time monitoring of consumption of electric energy can be applied by means of so-called smart meters. These meters are becoming more widely used for the real-time analysis of electric parameters. In addition to collecting data in real-time, they can also send data by means of wireless technologies to data processing centers, as well as being able to receive information and process it, i.e., they possess bidirectional communication technology [4].

Among the techniques for real-time monitoring of electric energy consumption, two can be highlighted. One is the technique denominated *Intrusive Load Monitoring* (*ILM*) and the other *Non-Intrusive Load Monitoring* (*NILM*).

The intrusive monitoring of loads based methods, requires a number of meters that is at least equal to the number of loads that will be monitored. This technique holds the advantage of one possessing, without errors, the individual consumption of each individual piece of electrical equipment or device being monitored. However, there exists the disadvantage of high cost, due to the quantity of meters and meter locations that need to be used and identified. Conversely, the technique of disaggregation by non-intrusive monitoring, needs only 1 measuring location. Further, techniques with disaggregation of loads manage to separate and classify which are being at a time instant. It is possible still, through the use of NILM, to derive what is known as the "load signature" [5].

As such, the loads can be divided into 4 groups [6]:

- Type I: Single state – On / Off – Examples: lamps, toasters, etc.
- Type II: Multi-State – Examples: washing machines, electric ovens, etc.
- Type III: Continually varying – Examples: drills and dimmable lamps.
- Type IV: Permanent consumption loads: televisions on stand-by, DVD appliances, etc.

Type 1 loads are easier to detect, but depending on their nominal power and consumption of electric energy, the technique applied for disaggregation might be unable to reach a conclusion as to which electrical equipment is being used at a specific time instant. Due to individual features pertaining to each electric equipment (current harmonics, power, characteristic of use, electromagnetic transients, among others), researchers have faced difficulties in disaggregating loads on a low power level. These difficulties are due to very similar features of operation, that are found in some equipment, as illustrated in Fig. 1.

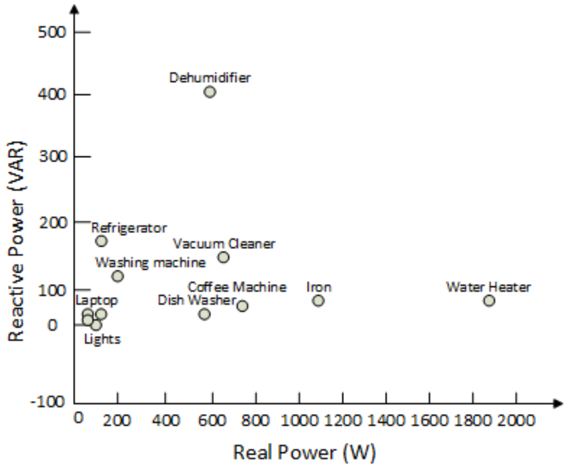

Fig. 1. Nominal power features of different appliances.

Among the techniques used for non-intrusive disaggregation of electric loads, the following can be highlighted:

- Optimization algorithms [7]-[8];
- Artificial neural networks [9]-[10];
- Hidden Markov Chains [11]-[12]
- Support Vector Machines [13]-[15].

Despite recent advances considering NILM techniques [2], there is no standard in regards to a solution be used, considering the state of the art. In this work, towards a thorough analysis of presented solutions, a comparative study of the state-of-the-art, considering artificial intelligence NILM based approaches, including electromagnetic transients, is presented. The end goal is to obtain, through statistical analysis of different evaluation metrics, considering synthetic and real-life data, a unbiased performance analysis, highlighting potential limitations and strengths of presented state of the art methods.

The main contributions of this study are:

- A comparative study between 3 (three) state-of-art artificial intelligence techniques for classifying disaggregation of residential electric loads through the non-intrusive monitoring method;
- The presenting of a convolutional neural network that results in higher rates of accuracy compared to previously published studies.

The remaining of the paper is divided as follows. Section II presents a background in machine learning solutions used for MILM. Section III presents details of the synthetic a real-life data used for the comparative study. Section IV presents a case study. Section V presents a comparative analysis with other studies. Section VI presents the main findings of this work.

II. MACHINE LEARNING SOLUTIONS

A. *Artificial Neural Networks*

In order to construct an artificial neural network, it is necessary to interconnect various artificial neurons, forming as such a type of graph. The format that these connections take determines that more commonly known as the architecture of the network.

In this paper, the network architecture known as Multilayer Perceptron (MLP) was used. In this type of network, all the neurons of each layer are connected to the neurons in the following layer, as shown in Fig. 2.

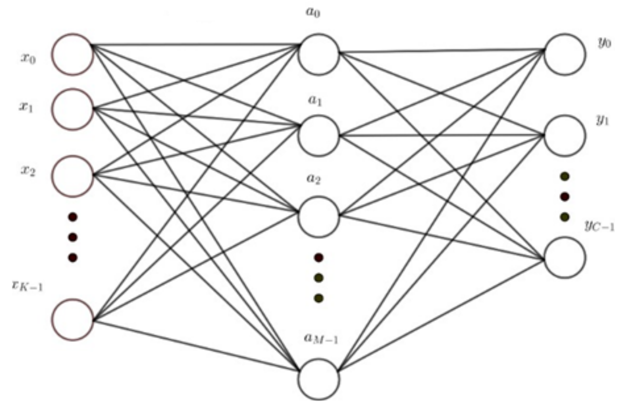

Fig. 2. Example of an MLP neural network

In figure 2, x_k represents the input data, y_c the output data and a_M the neurons from the hidden layers.

In order that the best results are reached for the classification proposed herein by means of the dense neural network, the Fourier transform was applied to the input signal, in order that features were extracted from the current signal used, and these employed in the training of the network.

B. *Long short-term memory network (LSTM)*

Recurrent Neural Network (RNN) has difficulties in learning long-term dependencies. To resolve this problem, the LSTM was designed. The LSTM neural network consists of LSTM units that contain three special gates, which are designed to control the flow of information inside each memory block [16]. The structure of the LSTM, denominated as cell, is seen in Fig. 3.

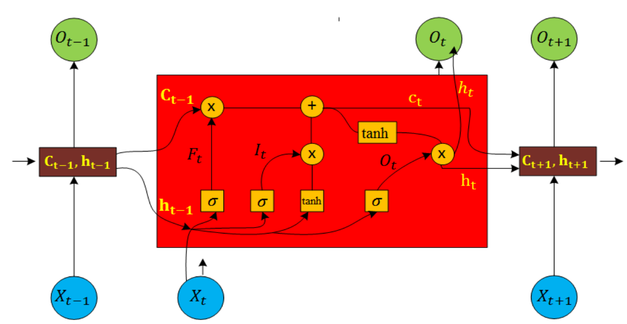

Fig. 3 Structure of the LSTM network.

The first gate is called the forget gate, F_t, which supplies a weight to the state of forgetfulness and has the function of deciding how much previous memory can be thrown out of the state cell. This can be calculated in the following way:

$$F_t = \sigma(W_F \times [h_{t-1}, X_t] + b_F) \quad (1)$$

Where $\sigma(\cdot)$ is a nonlinear elementwise function, usually called the activation function, W_F and b_F are the weights and forget gate matrix bias, respectively; h_{t-1} is the last moment hidden layer output and X_t is the actual input moment.

The next step consists of calculating the input value of gate I_t, which decides the amount of new information that will be stored in the state cell. This step has two parts: in the first, the sigmoid layer of the gate filters some information from h_{t-1}, X_t and C_{t-1}, after a Tanhyperbolic layer creates the value of the new candidates S_t:

$$i_t = \sigma(W_i \times [h_{t-1}, X_t] + b_i) \quad (2)$$

$$S_t = tanh(W_s \times [h_{t-1}, X_t] + b_s) \quad (3)$$

Where W_i, W_s, b_i and b_s are the corresponding weights and bias.

Therefore, the cell state C_t can be calculated as:

$$C_t = F_t \cdot C_{t-1} + i_t \cdot S_t \quad (4)$$

Finally, the output gate O_t is calculated to control the amount of information that will flow to the outside of the cell (5). After, the values of the cell are normalized between -1 and 1, by means of the Tanhyperbolic layer and then multiplied by the output sigmoid layer. The output is defined as (6):

$$O_t = \sigma(W_i \times [h_{t-1}, X_t] + b_i) \quad (5)$$

$$h_t = O_t \cdot tanh\, tanh\, (C_t) \quad (6)$$

C. Convolutional Neural Network (CNN)

The CNN is an artificial neural network able to learn features and classes, in it's different layers. Further this architecture allows the continuous adjustment of the parameters at running time, based on accuracy, giving more value to one layer than another, considering the problem at hand.

The CNN relies on the natural stationary property of an image, i.e., the statistics of one part of the image are considered the same as any other part ,and information extracted at one part can also be employed to other parts. Furthermore, deep CNNs usually obtain different levels of abstraction for the data, ranging from local low-level information in the initial layers (e.g., corners and edges), to more semantic descriptors, mid-level information (e.g., object parts) in intermediate layers and high-level information (e.g., whole objects) in the final layers.

CNNs are very similar to conventional neural networks, as the MLP network. These are constituted of neurons that have weights (or parameters) that "learn". Each neuron receives input and produces a scalar product, with the option of being followed by a nonlinear activation function.

In particular, the layers from a CNN possess neurons organized in three dimensions: width, height and depth (in this case, depth refers to the third dimension of an activation volume and not the depth of the network architecture, which is determined through the number of hidden layers).

A network with convolutional architecture is constituted of a sequence of layers, where each layer transforms an activation volume into another through a differentiable function. The three main layers used in the construction of the convolutional architecture are Convolutional Layer, Pooling Layer and Dense layer. Fig. 4 illustrates an example of convolutional architecture.

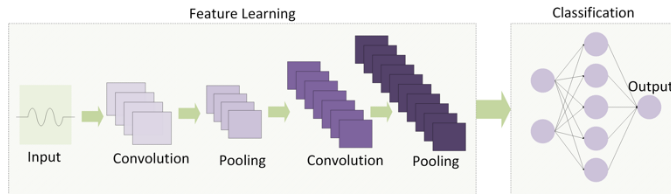

Fig. 4. Example of convolutional architecture

The function of each layer in the architecture can be summarized as:

- Convolutional layers: These possess the function of extracting the best features from the inputt signal in an automated fashion, without the need, for example, to perform signal filtering..
- Pooling layers: These are used to scale and map the data after convolution, thus reducing the data dimension, highlighting only important information.
- Batch Normalization Layers: These are used to improve speed, performance and stability of deep architectures (i.e., architectures that possess many hidden layers) during network training. Furthermore, these possess the function of reducing overfitting on training data.
- Dense output layer: This is responsible for attributing a class of input signal, and depends on the number of classes.

One of the benefits of using CNN architecture, is that electromagnetic transients can be detected by this classifier and learned, thus not interfering on its further detection and classification.

For this study, the CNN architecture used is presented in Fig. 5.

Fig. 5. Convolutional based architecture.

III. DATABASE

The data used during training, validation and testing of the models are referred to the database Reference Energy Disaggregation Dataset (REDD) [17]. This database contains information for the energy consumption of a whole house and of the circuit/device. In the version of the database used in this study, there exists information that refers to six residences. For each one, data was stored pertinent to apparent power Volt-Amps (VA) at the main circuit level of the residence, sampling at a rate of 1 Hz and the circuit/device level at a rate of 0.33 Hz. Each residence possesses two main circuits, denominated as *Main 1* and *Main 2*. However, for two of these six residences, signals were also captured for voltage and current at the main circuit of the residence, both sampled at a rate of 15 kHz.

In order to perform this study, the data referring to the current signals of electrical devices was used. Only data from the two residences was used, house number 3, as this possessed the highest quantity of data. As with power, there exist current signals for *Main 1* as well as for *Main 2*.

Electrical current data is sampled at a rate of 15 KHz, therefore, as explained by [17], data compression is used, in which the waveforms and data that indicate their timestamps are stored. The Timestamp is a type of data capable of storing information for the year, month, day, hour, minute and second. Through such, the files current1.dat and current2.dat with measurements added from the main circuits *Mains 1* and *2*, possess lines each with the following information:

1. A decimal timestamp value that allows for the fractional part;
2. A whole number, counter for the duration of the waveform cycles;
3. 275 real values, indicating the waveform value (in amperes), which are equally spaced within the cycle.

Fig. 6 presents the data measurement format for current available on the database REDD.

```
1297340206.597013 135.000000 0.000000 3.623859 7.254136 10.949398 ...
1297340208.844086 722.000000 0.000000 3.638527 7.249567 10.929027 ...
```

Fig. 6. Example of data contained in files from REDD.

A. Data pre-processing

The acquisition of test and validation data and construction of a database with this information is a complex task, which further involves privacy constraints. This task can be time consuming, thus, the option was made for using an already existing database (REDD). Nevertheless, it was still necessary to perform data pre-processing, in order that the data can in fact be used in the training of the classifiers.

The apparent power and current values of each measured circuit cannot be added directly, since each circuit possesses a different power factor. However, the nominal active powers (W) from electrical devices can be, obtaining as such, the level of active demand from the main circuit, in accordance with this sum.

During this step of the study, labelling was performed for current signals obtained by measuring the main circuits in the residence, based on the operational period of each device using the respective active power signal. Labelling consisted of assigning an integer to a whole number for a given current signal, based on the combination of active devices during the period where the signal was measured, in such a way that for each device combination found in the database, there exists a single whole number that identifies this combination.

This step was subdivided into 4 parts: identification of devices connected to the circuits; identification of the active period of devices; numbering of combinations and labelling of signals. Table I illustrates the post-processing information used, regarding the quantity labelled current signals.

TABLE I

INFORMATION REFERRING TO THE QUANTITY OF LABELLED CURRENT SIGNALS.

Description	***Main 1***	***Main 2***
Number of connected devices	12	8
Number of device combinations	187	64
Labelled current waveforms	858420	853040

In order to carry out evaluation of the state-of-the-art strategies used in NILM, researchers frequently make use of specific metrics. In this study, four (4) of these metrics were considered in order that the classifiers used herein could be assessed. These metrics are Recall (8), Precision (9), Accuracy (10) and F1-Score (11) (Liu, 2019):

$$Recall\ (R) = \frac{TP}{TP + FN} \tag{8}$$

$$Precision\ (P) = \frac{TP}{TP + FP} \tag{9}$$

$$Accuracy\ (ACC.) = \frac{TP + FN}{P + N} \tag{10}$$

$$F1 - Score = \frac{2 \times P \times R}{P + R} \tag{11}$$

Where, Recall is the rate of true positives or TP sensitivity (correctly predicted that the equipment was switched on), FP are the false positives (predicted device switched on, but was switched off), FN are the false negatives (device switched on, but predicted as switched off). Precision refers to the predicted positive values. Accuracy is the proportion of real results in all cases. F1-Score is the harmonic mean between Precision and Recall.

IV. RESULTS AND DISCUSSIONS

In this section, a case study is presented. Results analysis and discussed are further presented.

A. Comparative analysis

First, 858,420 examples for *Main 1* and 853,040 examples for *Main 2* were generated. Considering training of the ML-based solutions, presented previously, the 10-fold cross-validation method was used [ref]. 22.5% of the data were selected randomly for validation, 67.5% for training and 10% for testing. Table II summarizes the data allocation. Table III summarizes the structures of the models used.

TABLE II

DATA USED IN THIS STUDY

Data type	Main 1	Main 2
Train set	579,433	575,802
Validation set	193,145	191,934
Test set	85,842	85,304

TABLE III
STRUCTURE OF THE CLASSIFIERS FOR NILM.

Name of network	Number of layers (without including BN and Dropout layers)	Type
MLP	Layers: 2 Dense layers: 4 Activation function: ReLu	ANN
LSTM	Layer LSTM: 3, Unit number: 32 Activation function: tanh	RNN
CNN	Layer 1D conventional: 6 Dense layer: 3 Activation function: ReLu	CNN

The tests were performed on a PC with an Intel ®Core^{TM}i7, 3.40 GHz X 4, 16GB RAM processor and video card from ®NVIDIA GEFORCE GTX1080 8G GPU. The accuracy values and losses for training and validation are shown in Figs. 6 and 7, respectively.

At each epoch, the accuracy of the models in the validation set will be detected when it is improved. Thus, the models with the best performances in the validation and training set are selected as the final model. The final models and their properties are shown on Table IV and V.

The mean performances for Main are shown on Tables VI and VII. Table VIII shows the general average performance for residence 3. On all tables, the best results are in bold.

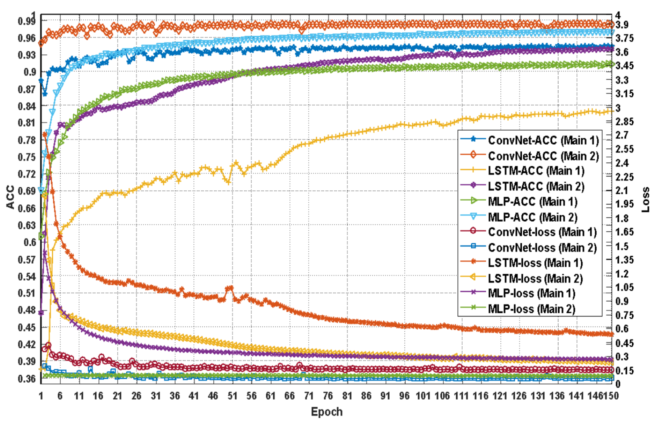

Fig. 7. Accuracy and loss values during training.

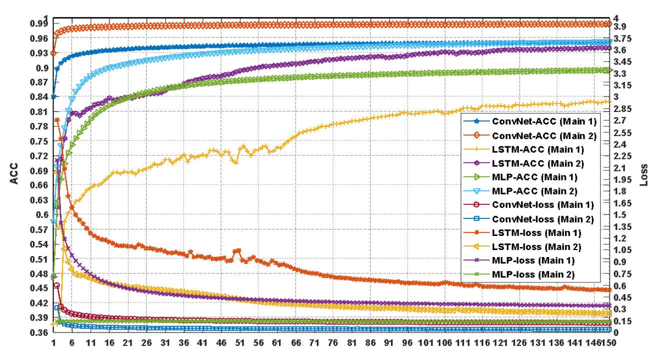

Fig. 8. Accuracy and loss values during validation.

TABLE IV
THE TRAINING MODELS AND THEIR PERFORMANCES.

Type of ANN	Highest loss in validation set		Best accuracy in validation set		Elapsed epoch	
	Main 1	Main 2	Main 1	Main 2	Main 1	Main 2
CNN	0.144	**0.054**	0.944	**0.984**	144	140
LSTM	0.526	0.235	0.834	0.940	142	138
MLP	0.263	0.086	0.913	0.970	149	149

TABLE V
THE TRAINING MODELS AND THEIR PERFORMANCES.

Type of ANN	Training time		Number of parameters	
	Main 1	Main 2	Main 1	Main 2
CNN	330 min	310 min	121,819	105,952
LSTM	1590 min	1560 min	19,227	15,168
MLP	55 min	48 min	34,171	30,112

TABLE VI
MEAN PERFORMANCE FOR MAIN 1

Main 1 - Average				
ANN	R	P	ACC	F1-Score
CNN	**96.37%**	**96.55%**	**99.40%**	**96.46%**
LSTM	76.51%	85.57%	96.67%	79.69%
MLP	94.58%	95.18%	99.18%	94.87%

TABLE VII

MEAN PERFORMANCE FOR MAIN 2

Main 2 - Average				
ANN	R	P	ACC	F1-Score
CNN	**98.24%**	**98.32%**	**99.78%**	**98.28%**
LSTM	81.27%	88.33%	98.14%	84.33%
MLP	97.69%	97.54%	99.63%	97.58%

TABLE VIII
GENERAL PERFORMANCE FOR RESIDENCE 3

General Average				
ANN	R	P	ACC	F1-Score
CNN	**97.30%**	**97.43%**	**99.59%**	**97.37%**
LSTM	78.89%	86.95%	97.40%	82.01%
MLP	96.13%	96.36%	99.40%	96.22%

Noteworthy here is that the indicators for *Main 1* were inferior when compared to those of *Main 2* for all the classifiers tested. This can be explained through the fact that some circuits, in light of the low quantity of circuits for classifier training, have difficulty in learning and recognizing these devices during the validation tests. However, for circuit *Main 2*, the lower quantity of devices for identification and the higher quantity of information from these were facilitating points, which allowed the classifiers to obtain better performance indexes.

From among the classifiers, one notes that LSTM obtained the worst performance results. CNN was the network with the best performance results. Despite the MLP neural network obtaining a performance very close to that of CNN, its disadvantage is found in the need for using prior feature extraction (load pattern, electromagnetic transients, among others) techniques on the signal analyzed, and only after can the MLP network be trained. This does not occur on CNN, one reason is that in its very own structure these characteristics are extracted in order to improve the performance of the network. In spite of this, due to its simpler structure, the MLP obtained the lowest time for training when compared to the others.

Further, among the techniques analyzed for classification considering NILM, the convolutional neural network CNN, was that which obtained the best performance indexes.

V. Comparative Analysis with Other Studies

As seen from Table IX, the convolutional network ConvNet presents a better performance over recently published methods. This comparison is performed only with studies that investigate NILM for residence 3 from the databank REDD. The small differences that can be found are due, as an example, to the fact that some authors use only values referring to one day, three days, or only some devices found in the databanks, while the classifiers tested herein use all devices. However, one notes that the accuracy of ConvNet was higher than that found for all cases researched in the literature. Furthermore, if one analyses only the devices that belong to circuit Main 2, the performance indicators are superior to all those encountered in other studies.

Table IX. Comparison with other studies.

Authors	Classifier	Manipulated Parameter	Feature extraction technique	R (%)	P (%)	ACC (%)	F1-Score (%)
Liu et. al. (2019) [18]	FCAM	Power	Significance Threshold	95.58	97.74	97.90	97.83
Tabatabaei, Dick and Xu (2017) [19]	MLCA	Power	Wavelet Transform	-	-	-	95.90
Kong et al. (2018) [20]	HMM	Power	Iterative k-means	-	-	83.50	-
Bhotto, Makonin and Bajic (2016) [21]	Aided Linear Integer Programming	Power	None	-	-	96.00	-
ConvNet general	CNN	Current	Auto	**97.30**	97.43	**99.59**	97.37
ConvNet Main 2	CNN	Current	Auto	**98.24**	**98.22**	**99.78**	**98.28**

*FCMA – Fuzzy Multi-Clustering Algorithm; MLCA – Multi-La

Moreover, the automatic selection of features performed by the CNN can improve the classification accuracy on the NILM, since the CNN extracts the most critical features from the different combinations of devices analyzed.

Finally, it becomes evident that the use of electric current signals to those of power load more features from the signals, this fact was crucial toward improving the performance of the classifier when compared to previous studies.

VI. Conclusion

This paper presented a comparative study of ML-based solutions of NILM methods. Three solutions were presented and analyzed, a MLP, LSTM and CNN based solutions. NILM has as one important goal the disaggregation of electric loads. For such, the ML-based solutions are put forward for the use of electric current data instead of power for the training and learning of classifiers. To this end, a labelling methodology was used. Three classification techniques that use state-of-the-art artificial intelligence were tested: an MLP network, a CNN and a recurrent network of the LSTM type.

The results show that the CNN proves to have best performance indexes for use in cases of NILM. Further, since the CNN based solution possesses, in its infrastructure, filters that are adaptable to the automatic extraction of signal features (as load behavior, electromagnetic transients, among others), this decreases much the cost of data pre-processing.

VII. Acknowledgment

This study was realized with support in part from the National Council for Scientific and Technological Development – CNPq Project 409687/2018-9 and in part from Mato Grosso Support Foundation - FAPEMAT Project 204690/2017.

VIII. References

[1] PROCEL –Programa de Conservação de Energia Elétrica. Resultados PROCEL 2017. Available in <http://www.procelinfo.com.br/resultadosprocel2017/docs/rel_procel2017_web.pdf>.

[2] Zoha, A.; Gluhak, A.; Imram, A. M.; Rajasegarar, S. Non-intrusive load monitoring approaches for disaggregated energy sensing: A survey. Sensors, Vol. 12, pp 16838-16866, 2012.

[3] Ehrhardt-Martinez, K.; Donnelly, K. A.; Laitner, J. A. Advanced Metering Initiatives and Residential Feedback Programs: A Meta-Review for Household Electricity-Saving Opportunities. Technical Report E105 for American Council for an Energy-Efficient Economy (ACEE): Washington, DC, USA, 2010.

[4] Buchholz, M. B.; Styczynski, Z. Smart grids – Fundamentals and technologies in electricity networks. Springer e-book, 412 p., 2014.

[5] Zeifman, M.; Roth, K. Nonintrusive appliance load monitoring: review and outlook. IEEE Conference on Consumer Electronics – ICCE. 2011.

[6] Hoyo-Montaño, J. A.; Pereyda-Pierre, C. A.; Tarín-Fontes, J. M.; Leon-Ortega, J. N. Overview of non-intrusive load monitoring. International Conference on Power Electronics – CIEP. 2016.

[7] Liang, J.; Ng, S.K.K.; Kendall, G.; Cheng, J.W.M. Load signature study Part I: Basic concept, structure, and methodology. IEEE Trans. Power Del. 2010, 25, 551–560.

[8] Baranski, M.; Voss, J. Genetic Algorithm for Pattern Detection in NIALM Systems. In Proceedings of IEEE International Conference on Systems Man and Cybernetics, Hague, The Netherlands, 10–13 October 2004; Volume 4, pp. 3462–3468.

[9] Ruzzelli, A.G.; Nicolas, C.; Schoofs, A.; O'Hare, G.M.P. Real-Time Recognition and Profiling of Appliances through a Single Electricity Sensor. In Proceedings of the 7th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks, Boston, MA, USA, 21–25 June 2010; pp. 1–9.

[10] Chan, W.; So, A.; Lai, L. Wavelet Feature Vectors for Neural Network Based Harmonics Load Recognition. In Proceedings of International Conference on Advances in Power System Control, Operation and Management, Hong Kong, October 2000; Volume 2, pp. 511–516.

[11] Zeifman, M. Disaggregation of home energy display data using probabilistic approach. IEEE Trans. Consum. Electron. 2012, 58, 23–31.

[12] Lin, G. Y.; Lee, S.C.; Hsu, Y.J.; Jih, W.R. Applying Power Meters for Appliance Recognition on the Electric Panel. In Proceedings of the 5th IEEE Conference on Industrial Electronics and Applications, Melbourne, Australia, 15–17 June 2010; pp. 2254–2259.

[13] Patel, S.N.; Robertson, T.; Kientz, J.A.; Reynolds, M.S.; Abowd, G.D. At the Flick of a Switch: Detecting and Classifying Unique Electrical Events on the Residential Power Line In Proceedings of the 9th International Conference on Ubiquitous Computing, Innsbruck, Austria, 16–19 September 2007; pp. 271–288.

[14] Srinivasan, D.; Ng, W.; Liew, A. Neural-network-based signature recognition for harmonic source identification. IEEE Trans. Power Del. 2006, 21, 398–405.

[15] Kato, T.; Cho, H.S.; Lee, D. Appliance Recognition from Electric Current Signals for Information-Energy Integrated Network in Home Environments. In Proceedings of the 7th International Conference on Smart Homes and Health Telematics, Tours, France, 1–3 July 2009; Volume 5597, pp. 150–157.

[16] Hochreiter, S.; Schmidhuber, J. Long short-term memory. Neural Comput. 1997; 9(8):1735-80.

[17] Johnson, Z. K.. M. J. Redd: A public data set for energy disaggregation research. In proceedings of the SustKDD workshop on Data Mining Applications in Sustainability, p. 6, 2011.

[18] Liu, Q., Kamoto, K. M., Liu, X., Sun, M., & Linge, N. (2019). Low-Complexity Non-Intrusive Load Monitoring Using Unsupervised Learning and Generalized Appliance Models. IEEE Transactions on Consumer Electronics, 65(1), 28–37. https://doi.org/10.1109/TCE.2019.2891160.

[19] Tabatabaei, S. M., Dick, S., & Xu, W. (2017). Toward Non-Intrusive Load Monitoring via Multi-Label Classification. IEEE Transactions on Smart Grid, 8(1), 26–40. https://doi.org/10.1109/TSG.2016.2584581.

[20] Kong, W., Dong, Z. Y., Ma, J., Hill, D. J., Zhao, J., & Luo, F. (2018). An Extensible Approach for Non-Intrusive Load Disaggregation with Smart Meter Data. IEEE Transactions on Smart Grid, 9(4), 3362–3372. https://doi.org/10.1109/TSG.2016.2631238.

[21] Bhotto, Md. Z. A., Makonin, S., Bajic, V. Load Disaggregation Based on Aided Linear Integer Programming. IEEE Transactions on Circuits and Systems – II: Express Briefs, 64(7), 792 – 796. https://doi.org/10.1109/TCSII.2016.2603479