Syncnet network application for identification of electrical charges

Jorge J. F. Filho[1], Marconi J.H. Magnani[1], Thyago R. Souza[1], Daniel C. Cavalieri [1], Cassius Z. Resende[1]

[1]*Graduate Program in Automation and Control, Federal Institute of Espírito Santo*
ES-010, Km-6,5, Manguinhos, 29173-087, Serra/Espírito Santo, Brazil
jorge@in9automacao.com.br,marconi@in9automacao.com.br,thyago@in9automacao.com.br

Abstract. The search for optimization of energy resources is growing every day, whether for environmental, financial or economic reasons. In industries and homes, knowing which equipment is turned on, when it is turned on, how often it occurs and how long it is used is essential information for energy optimization algorithms. Therefore, recognizing the characteristic signature of the equipment is a big step. This work proposes the use of a convolutional network using a Sincnet layer initially proposed by Mirco Ravanelli and Yoshua Bengio, used for human speech recognition, the dataset used is the WHITED which has the signature of several loads of different devices, this work will Evidencing the tests carried out on isolated loads and their identification, the success of this work will allow this network to carry out tests of coupled loads in the future and, in the future, integrated to an energy optimization system, monitoring a single point in the electrical network.

Keywords: Energy, Sincnet, Load Identification.

1 Introduction

Currently, there is a search to optimize the use of energy, also aiming to reduce energies from fossil sources, counteracting the current increase in energy demand caused by increased technologies and population growth. As the search for energy optimization advances, the load identification research advances together, as it enables the identification of inefficient loads, enables the use of loads at opportune times, improvements in power factor correction algorithms, etc. The gains and the possibility of application are great, it is estimated that up to 33% of all energy consumed in commercial buildings is consumed by devices that use plugs. [1] For these cases, it is expected that, on average, energy consumption will increase by 2.6% for the period 2019 to 2050. [2]

One of the methods for acquiring energy consumption data is NILM (Non-intrusive Load Monitoring), this method consists of installing an energy meter in the main supply, this meter starts to monitor the energy consumption of each connected load in a given construction, the NILM method has the advantage of not generating disturbance in the system during its operation [1][3][4].

There are already available in academia and on the internet some datasets that use NILM methods for data acquisition and thus formulate reliable databases with loads of various equipment, one of these datasets is the WHITED (Worldwide Household and Industry Transient Energy Dataset), a dataset that includes a variety of equipment, the variables were measured in 16 bits and with sampling of 44.1 kHz, the dataset consists of 10 sampling of each electrical device [5]. This work proposes, using the WHITED dataset, an intelligent system to identify electrical charge signatures using deep neural networks.

Neural networks are tools that are used in the most diverse applications, covering the most diverse areas. Examples in medicine can be cited in which it is used to determine kidney diseases by analyzing the characteristics of the organ.[6] The Deep Learning tools, emerged and widely developed in recent years, which is a multilayer network normally trained in a supervised way and uses the error backpropagation, in the training stage it uses a large amount of labeled data, but its results are considered impressive in some certain conditions surpassing the results obtained by human beings as exemplified by Tavanaei, Ghodrati, Kheradpisheh, Masquelier and Maida [7]. There is also a highlight for Ravanelli and Yoshua Bengio [8], who developed Sincnet, a neural network that uses several layers with different structures to extract signal characteristics and using them as a voice identification technique, which will be adapted in this work for the recognition of electrical charges. Currently, there are studies using conventional neural and convolutional network techniques for charge identification using the current and voltage trajectory method as performed by Leen De Baets, Joeri Ruyssinck, Chris Develder, Tom Dhaene and Dirk Deschrijver [9].

2 Dataset Whited

The Whited dataset was developed by Kahl, M., Haq, AU, Kriechbaumer, T. & Jacobsen, H.-A [5], the idea is to measure electrical characteristics of various devices using a low cost sound card, the data were collected in homes and small industries, data were obtained after measurements of 5 seconds after energizing the devices, a total of 10 measurements were performed for each device. The technique used in this capture is a non-intrusive technique, that is, the measurement process does not affect the measured variable, this method is called NILM, there are several datasets performed with this technique, including REDD, BLUED, PLAID and HFED .[5]

Figure 1 shows the measurement device used to generate the dataset, it consists of an AC-AC transformer and a set of sockets where the load is turned on, it is also possible to recognize the sound acquisition card used to capture the data due. The Whited dataset has signals recorded at a frequency of 44.1 kHz and data is recorded in 16 bits. All recorded data were processed in MATLAB to perform a standard 5 second record including a 100ms blank at the beginning of each measurement, the data were processed to capture the disturbance of the current caused by turning on an electrical device, the implemented algorithm also makes the correction of each meter's calibration factor to record the actual value.

Figure 1. Measuring device. Source: [5]

Figure 2 shows an example of data present in the dataset for an electrical charge, this data is saved in FLAC format, which is commonly used as a format for recording sound data, this format uses a compression method without promoting frequency cuts or by sampling rate, a way that maintains the original characteristics of the measured data.

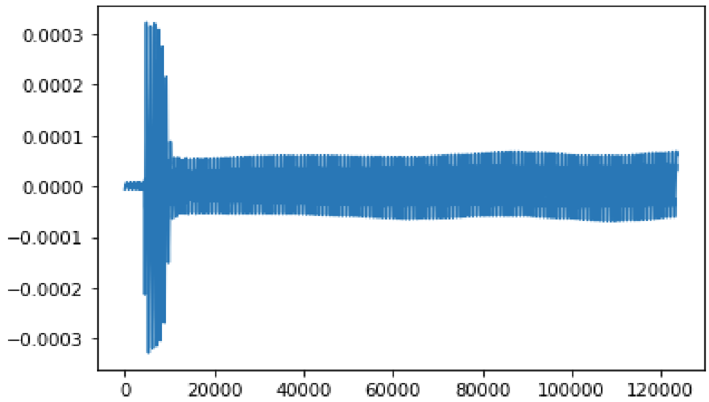


Figure 2. Result of measurement of a load, element present in the dataset. Source: [5]

3 Sincnet Network

Artificial Neural Networks are systems composed of simple processing units called neurons, which perform certain mathematical functions, according to Braga, Ludemir and Carvalho [10]. In this present work we use some layers of neurons that work in different ways, one of these layers is comprised of convolutional networks, this network has been shown to be very efficient in image processing applications. For Ponti [11], deep learning methods are now the state of the art in many machine learning problems, in particular in classification problems. Figure 3 shows how a convolutional network works, the network using windows captures segments of the main image and through error corrections each layer works as self-adjusting filters, each filter has a weight matrix for extracting various visual characteristics, these characteristics are stored in the training phase, and so these filters are used in the future for predictions.

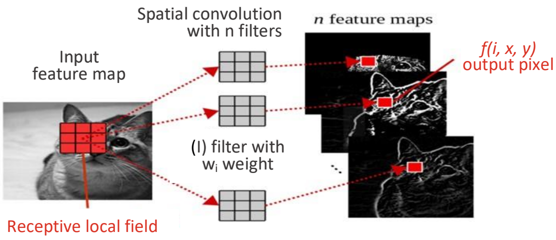


Figure 3. Spatial convolution. Source: Ponti (2017)(adapted)[11]

In Figure 4, it is possible to observe the application of two convolution layers in an RGB image where in the first layer 5x5x3 filters are applied as a result there are 4 feature maps these resulting layers go through another convolutional layer with 3x3x4 filters generating new feature maps [11].

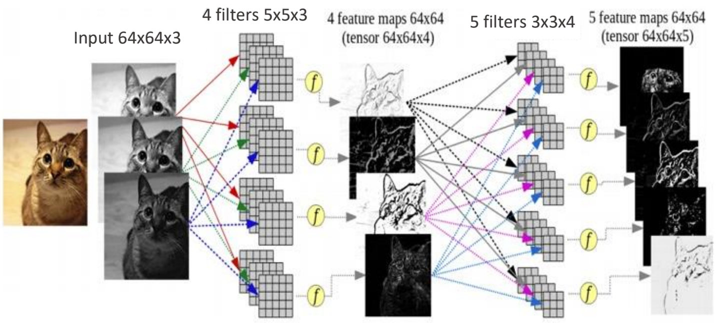


Figure 4. Network with two convolutional layers. Source: Ponti (2017) (adapted)[11]

The Sincnet network uses convolutional layers and dense layers for classification, but there is an additional layer at the beginning of the network which is a specific layer that works by extracting characteristics of a sound signal, these characteristics are extracted through the use of low pass filters , high-pass and band-pass, these filters are generated for each sound signal segment, these portions are extracted from the original signal and are somehow superimposed in small bands, each piece is extracted as in a moving window that moves along the original signal forming the segments that will be forwarded to the network input, these signals then pass on the first layer where several filters are created in order to create a feature map in each spectrum band, these feature map are forwarded to the layers of convolutional networks that will extract more features, and finally these features go through layers of dense networks as to promote the classification of each analyzed sound. In this network, it was used to identify a speaker through the analysis of the generated sound, the dataset used in the development of this network is TIMIT, which has several sounds recorded in WAV format, which is an audio format that preserves the sound characteristics as much as possible [8],

The Figure 5 shows the architecture proposed by Ravanelli and Bengio. In the proposed architecture the voice waveform passes through a series of filters, these filters operate with segments of the original wave, making then a convolution by defined functions, these defined functions are groups of rectangular bandpass filters, the training of this layer of the network is based on finding the two frequencies of each bandpass filter that extract the representative characteristics of each signal [8].

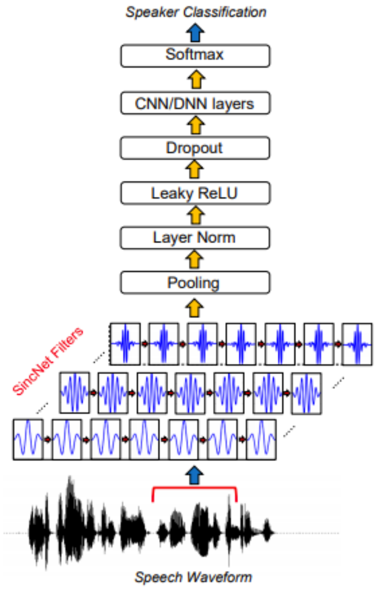


Figure 5. Network proposed by Ravanelli and Bengio. Source: Mirco Ravanelli and Yoshua Bengio (2018)[8]

The final architecture proposed (Figure 5) was then an input sicnet layer with 80 of 251 filters on the input, passing through a maxpooling of 3, the maxpooling downsampling the data using the maximum value, a layernormalization to normalize the activations of the previous leayer, a leakyrelu that is a layer with a leakyrelu activations this activation is based on relu activation but has a small slope for negative values, a convolutional 60x5x1, another maxpooling of 3, another layernormalization, another leakyrelu, a flatten, a dense layer of 2048, a batchnoramlization that normalize batches, another leakyrelu, another dense of 2048, another bachnormalization, another leakyrelu, and then the last dense layer of 2048, the last bachnormalization, and the last leakyrelu, and finally at the output a dense of 126 for every class he trained [8]. This architecture looks like a classical convolutional network, the significative change made is the first layer which is modified to a sincnet layer which is a custom layer specialized in signal filtering. The Sincnet networks have advantages over networks that do not use filter layers, as there are networks that filter using a fully convolutional layer, but these networks tend to have many more parameters than sincnet as it grows linearly in a geometric progression and it also has a high start size compared to sincnet, which already has a reduced size and has an arithmetic progression, another advantage that sincnet has is that it has a faster convergence as the initial layer filters tend to respond faster in the phase training and this causes a faster response in training and prediction [8].

4 Development

The network was developed in Python language. Tensor Flow, Keras, Pandas, Numpy and Pysoundfile libraries were used in the Google desktop called Google Colab, some processing was done in Excel and VBA programming for basic data processing. As previously mentioned, the Whited dataset was used in this work, some processing was necessary in this dataset so that we could use it in the proposed network, the processes performed were as follows: classification in their respective classes, separation of data for training, validation and tests, normalization of the data, and finally removing the 100ms of null signal added at the beginning of each record.

The data was separated between the training, validation and testing data, in this sample we used a 7:2:1

division, being for every 10 files of the same class of the dataset 7 were intended for training 2 were intended for validation and one was intended for future testing. The test data is not part of the database used in the training and validation of the network, so it is data never presented to the network before, and it will be this data that will be used to verify the effectiveness of the network. To perform the normalization of dataset data, the minmax_scale function was used because it causes the data to be normalized between 0 and 1, since the dataset presents raw data, there is a lot of variability between the magnitudes of the data, in addition to the fact that they are transiting between negative and positive values, another important point in the database data is that the data in many cases have very low magnitudes, that is, values very close to zero and some values to zero, this implied a loss of convergence several times in the network because it caused several weights to zeros in some infinite cases, in addition to being a good practice, normalization in this case was extremely necessary due to their nature.

It is also observed that the emptiness or silence inserted intentionally present in each sample of the database was removed, at this stage most of the silence was removed, as a default 100ms of silence was inserted with a sampling of 44.1kHz, we have 4410 samples with value zero or close to it, so much of the value was removed because it was not necessary, as you can see in the image above, 410 points were still preserved at the beginning of the measurement with null value.

4.1 Network architecture

Figure 7 shows the structure of the network that was developed, the convolutional base of the network used is close to the VGG16 model but the input layer is a Sincnet layer, another important detail is that all convolutional layers are one-dimensional layers, dropout layers was added to prevent the network from having a greater degree of generalization, for the classifier two dense stacks were used - batch normalization - leaky relu in sequence and finally an output layer with the 55 classes that make up the database.

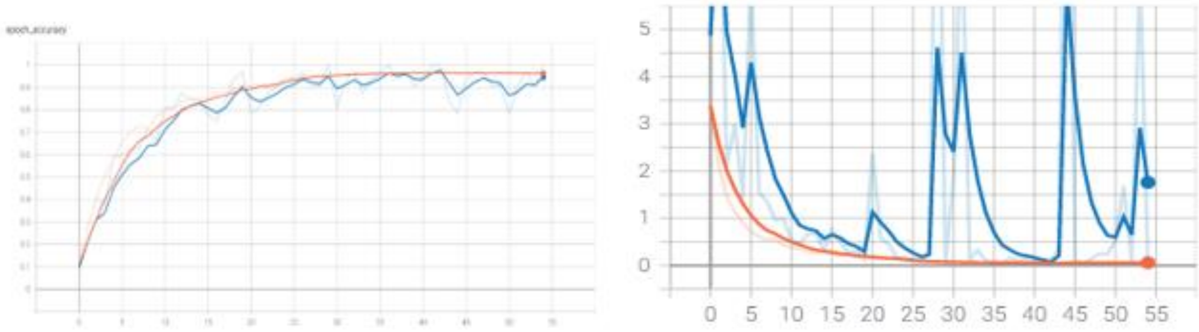
Layer (type)	Output Shape	Param #
sinc_conv_fast_1 (SincConvFa	(None, 4160, 200)	525
conv1d (Conv1D)	(None, 4160, 256)	153856
max_pooling1d (MaxPooling1D)	(None, 1386, 256)	0
dropout (Dropout)	(None, 1386, 256)	0
layer_norm (LayerNorm)	(None, 1386, 256)	512
leaky_re_lu (LeakyReLU)	(None, 1386, 256)	0
conv1d_1 (Conv1D)	(None, 1386, 128)	98432
conv1d_2 (Conv1D)	(None, 1386, 128)	49280
max_pooling1d_1 (MaxPooling1	(None, 462, 128)	0
dropout_1 (Dropout)	(None, 462, 128)	0
layer_norm_1 (LayerNorm)	(None, 462, 128)	256
leaky_re_lu_1 (LeakyReLU)	(None, 462, 128)	0
flatten (Flatten)	(None, 59136)	0
fc1 (Dense)	(None, 512)	30278144
batch_normalization (BatchNo	(None, 512)	2048
leaky_re_lu_2 (LeakyReLU)	(None, 512)	0
fc2 (Dense)	(None, 256)	131328
batch_normalization_1 (Batch	(None, 256)	1024
leaky_re_lu_3 (LeakyReLU)	(None, 256)	0
output (Dense)	(None, 55)	14135

Figure 6. Network architecture with its respective formats

4.2 Training

For the training it were used the optimizer Adam, the reduce learn rate on plateau and early stopping methods

The best result was obtained at time 43, with val_loss of 0.0227 and acc of 1.0, each batch was set at 200. In the graphs below, we can see the training evolution, the accuracy evolved until reaching the maximum level while the loss value was changing throughout the training and the reduce learn and early stopping techniques were necessary to reach a good result as shown in the Figure 7 in (a) it´s possible to understand how the accuracy progressed, and in (b) it´s possible to understand how the loss progressed, showing that early stop worked getting parameters at the epoch 43.



(a) (b)

Figure 7. Accuracy throughout training and loss throughout training

4.3 Results

One approach used to analyze the results is the confusion matrix as shown in the Figure 8, as we can see below, a class-by-class matrix was made in order to identify where there were the best and worst results. In this case, some results were selected that stood out for having good results and not satisfactory.

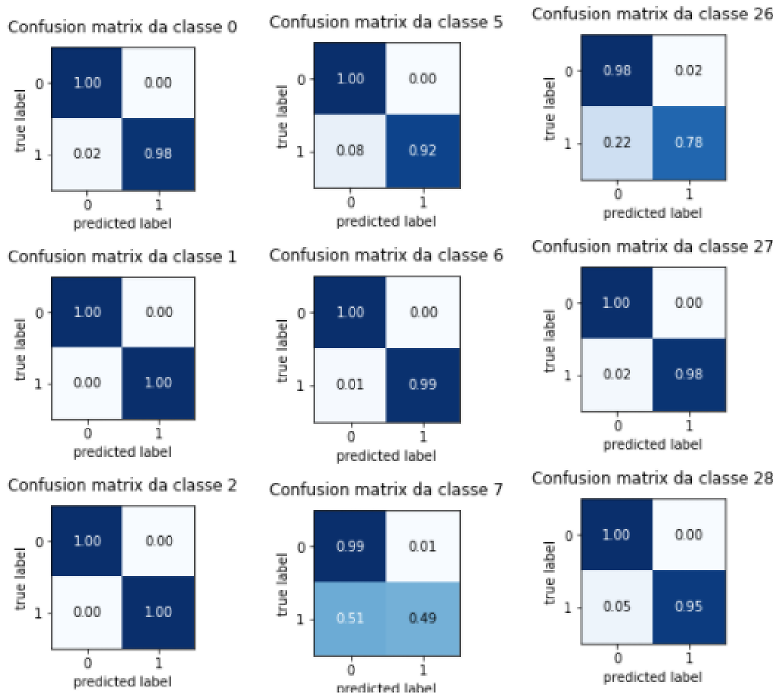


Figure 8. Result of some confusion matrices by class

As one can see, the results of some classes have a lot of false negatives, but of the 55 classes only 5 classes showed results well below the average, as we can highlight class 7 that had the worst result in the analysis of the confusion matrix, it must certainly be deepen the analysis in relation to the data collected, there may be failures in its collection or classification.

To validate the results, another approach was used using the F1 score index, whose result was 92% and 94% recall and 92% precision, these results translate into an acceptable performance.

5 Conclusions

This work presented an intelligent system to identify electrical charge signatures using deep neural networks. The chosen neural network was the Sincnet network, which uses frequency filters in the first layers, which are then connected to convolutional layers and dense layers to extract characteristics from each signature. An important point to be highlighted about the proposed network is the adjustment of the number of layers, as well as the number of neurons and filters applied, all these requirements were made aiming at high speed associated with high performance. Regarding the final results, several techniques were used to analyze the performance results and the values presented fit into networks of satisfactory performance, although there are difficulties in detecting certain classes such as the coffeemaker, kettle and toaster classes, the coffeemaker with the worst result of 49% correctness followed by toaster with 54% correctness, these items have a very resistive signature compared to the other components, these items also had poor results in other tests performed previously as presented in the work by De Baets[9] where these classes also contributed negatively to the performance of the neural network, for this it is recommended to increase the dataset used with more samples and balance the load data to verify the responses of neural networks and Sincnet in these specific points.

Acknowledgment. First and foremost, the authors would like to thank IFES and especially the Serra campus team.

Authorship statement. The authors hereby confirm that they are the sole liable persons responsible for the authorship of this work, and that all material that has been herein included as part of the present paper is either the property (and authorship) of the authors, or has the permission of the owners to be included here.

References

[1] Tekler, Zeynep Duygu & Low, Raymond & Zhou, Yuren & Yuen, Chau & Blessing, Lucienne & Spanos, Costas, 2020. "Near-real-time plug load identification using low-frequency power data in office spaces: Experiments and applications," Applied Energy, Elsevier, vol. 275(C).

[2] U. E. I. Administration, Annual energy outlook 2020: Commercial sector key indicators and consumption (2020).

[3] Abubakar, S. Khalid, M. Mustafa, H. Shareef, M. Mustapha, Application of load monitoring in appliances' energy management {a review, Renewable and Sustainable Energy Reviews 67 (2017) 235{245.

[4] S. Welikala, N. Thelasingha, M. Akram, P. B. Ekanayake, R. I. Godaliyadda, J. B. Ekanayake, Implementation of a robust real-time nonintrusive load monitoring solution, Applied energy 238 (2019) 1519{1529.

[5] Kahl, M., Haq, A. U., Kriechbaumer, T. & Jacobsen, H.-A. Whited-a worldwide household and industry transient energy data set. Proc. 3rd international workshop on non-intrusive load monitoring (2016).

[6] Raju, P., Rao, V. M., & Rao, B. P. (2018). Grey Wolf Optimization-Based Artificial Neural Network for Classification of Kidney Images. Journal of Circuits, Systems and Computers, 27(14), 1–21. https://doi.org/10.1142/S0218126618502316.

[7] Tavanaei, A., Ghodrati, M., Kheradpisheh, S. R., Masquelier, T., & Maida, A. (2019). Deep learning in spiking neural networks. Neural Networks, 111, 47–63. https://doi.org/10.1016/j.neunet.2018.12.002.

[8] Ravanelli, M. and Yoshua Bengio. "Speaker Recognition from Raw Waveform with SincNet." *2018 IEEE Spoken Language Technology Workshop (SLT)* (2018): 1021-1028.

[9] De Baets, L.; Ruyssinck, J.; Develder, C.; Dhaene, T.; Deschrijver, D. Appliance classification using VI trajectories and convolutionalneural networks. Energ. Build. 2018,158, 32–36, doi:10.1016/j.enbuild.2017.09.087

[10] BRAGA, A. P.; CARVALHO, A. C. P. L. F.; LUDEMIR, T. B. Redes Neurais Artificiais: teoria e aplicações. 1 ed. Rio de Janeiro: LTC, 2000.

[11] Ponti MA, Costa GBP da. Como funciona o deep learning [Internet]. In: Tópicos em gerenciamento de dados e informações 2017. Uberlândia: SBC; 2017. Available from: http://sbbd.org.br/2017/wpcontent/uploads/sites/3/2017/10/topicos-em-gerenciamento-de-dados-e-informacoes-2017.pdf