Monthly Energy Consumption Forecast: A Deep Learning Approach

Rodrigo F. Berriel*†, André Teixeira Lopes*, Alexandre Rodrigues, Flávio Miguel Varejão, Thiago Oliveira-Santos
Universidade Federal do Espírito Santo, Brazil
†Email: rfberriel@inf.ufes.br

***Abstract*—Every year, energy consumption grows world widely. Therefore, power companies need to investigate models to better forecast and plan the energy use. One approach to address this problem is the estimation of energy consumption in the customer level. Energy consumption forecasting problem is a time series regression task. It consists of predicting the energy consumption for the next month given a finite history of a customer. Machine learning techniques have shown promising results in a variety of problems including time series and regression problems. Part of these promising results are attributed to deep neural networks. Although investigated in other domains, deep architectures have not been used to address the energy consumption prediction problem. In this work, we propose a system to predict monthly energy consumption using deep learning techniques. Three deep learning models were studied: Deep Fully Connected, Convolutional and Long Short-Term Memory Neural Networks. Due to the sensitivity of these models to the input range, normalization techniques were also investigated. The proposed system was validated with real data of almost a million customers (resulting in over 9 million samples). Results showed that our system can predict monthly energy consumption with an absolute error of 31.83 kWh and a relative error of 17.29%.**

I. Introduction

Every year, energy consumption grows world widely [1], specially in developing countries like Brazil. The growth in energy consumption increases the need for better planning of energy use, which also includes the better planning of energy distribution and energy consumption measuring. Individual energy consumption prediction can help optimizing energy distribution and can also make the process of reading electrical meters less prone to human errors.

In Brazil for example, the process of measuring energy consumption is still very human dependent and due to law requirements has to be performed every month for each customer. To fulfill such requirements, power companies have several human personnel, i.e. readers, that goes monthly to each customer (residential or business customer) in order to read the electrical meter and to issue the bill for the consumption of that month. Each reader takes along a device that enables entering the consumption value, taking pictures of the meter and printing the bill *in loco*. Nevertheless, humans are affected by tiredness, by the difficult of reading electrical meters due to natural wear of the meter, by the lack of meter standardization, and other factors. Therefore, the fact of having humans to read the numbers in the meter can lead to a wrong billing. These billing errors increase the costs of the power companies, lower the customer reliability in the company and create problems with the energy regulatory agencies.

To hinder such problems, power companies try to predict the expected consumption of each customer in order to confront it with the value of the human reader. This requires two steps, one for predicting the value, and another for defining a tolerance around the predicted value. The combination of these two steps establishes an upper limit for the values to be read. If the read value is close to the predicted value (i.e. is less than the predicted value plus a tolerance), the bill is printed and handed to the customer directly. However, when the two values diverge (i.e. is larger than the predicted value plus a tolerance), a photo of the electrical meter is sent to an analyst that will later decide on the correctness of the reading. Finally, the monthly consumption value of each customer is stored in a log for general use.

This process can reduce the number of wrongly issued bills, but can also lead to problems if the predicted value or the tolerance are not very accurate. In this context, a bill can lead to four scenarios: the read value is wrong and it is larger than the predicted value plus the tolerance (true negative), the read value is correct and it is also less than the predicted value plus the tolerance (true positive), the read value is wrong and it is also less than the predicted value plus the tolerance (false positive), the read value is correct and it is larger than the predicted value plus the tolerance (false negative). Accurate predicted values and tolerances lead to the first two scenarios, whereas inaccurate predicted values and tolerances lead to the two last scenarios and also lead to the increase of costs. False positives lower company credibility and increase costs because customers might sue the company for charging more than it was consumed. False negatives increase costs because the power company needs to hire analysts to handle the bills and needs to pay for the posting costs of the bill.

Some power companies in Brazil use simple averaging and standard deviation of each customer to predict consumed value and to define the tolerance respectively. In fact, the bill is printed if the read value is less than a threshold. The threshold is given by the average plus k times the standard deviation considering the consumption of the past 12 months of a customer respectively, where k is an amplifier factor that was empirically defined by the power company as $k = 6$. Because this threshold can lead to a high number of bills to

*These authors contributed equally to this work

978-1-5090-6182-2/17/$31.00 ©2017 IEEE

be analyzed, the power companies usually print all bills below certain values depending on the type of the customer.

Although the methodology presented above comprises two steps, i.e. estimation of the consumed value (mainly performed by the average of the past 12 months) and of the tolerance (mainly defined by the standard deviation of the past 12 months), in this paper, focus is given to the first part only. More specifically, this work focus on finding a better regressor for the consumed value with the use of deep neural networks.

Neural network is a well-established machine learning technique and have been successfully applied to a large set of daily life problems, such as speech recognition, image classification, video analysis, prediction in stock market, weather forecast, and also prediction of energy consumption [2]. Lately, neural networks with deep architectures have gained attention of the research community due to their ability of capturing data behavior when considering large amounts of data. Two types of deep neural networks suit the problem addressed in this paper: Convolutional Neural Network (CNN) [3], [4] and Long Short-Term Memory (LSTM) [5]. The first is known to work well with images, but has also shown to be successful with temporal data [6]. The second network is based on recurrent networks and was designed to work with temporal data. Neural networks can address both types of problems, classification and regression, and therefore, can be directly applied as regressors for the first part of our problem, i.e. energy prediction.

In this work, we show a practical application of deep neural networks to the problem of monthly energy consumption forecast. This is a key issue for power companies and a better regressor could greatly improve the quality of their service. To propose the best neural network based solution, we compare the performance of the application with different types of neural networks, of architectures, of data normalizations, and of metadata. The study is performed on real data comprising monthly consumption of almost a million of customers in Brazil for a period of 24 months. The validation measures the error between the predicted value and the real consumption. Results showed that the proposed solution outperforms the baseline method described above. In addition, we also present an analysis of the different combination of methods proposed for the regression problem, and results show that LSTM networks can better handle this problem.

II. Background and Related Work

Machine learning has been applied to a large set of problems in our society. Indeed, a lot of daily use applications (like search mechanisms, text editors and personal assistants) have, in some aspect, an intelligent behavior allowed by machine learning algorithms. These algorithms are able to discover patterns in data, specifically when a huge amount of samples is available. There are already works in the literature that apply machine learning to the energy consumption prediction problem. Some of these works and others related to the proposed work are highlighted here. A more in-depth review of the energy consumption forecasting research field was presented by Zhao and Magoulès in [7].

Williams and Gomez [8] conducted a study with statistical learning methods to predict energy consumption. The authors employ three methods for forecasting the next month energy consumption: Linear Regression, Regression Trees and Multivariate Adaptive Regression Spline (MARS). The data used by the authors comprise consumptions from over $426,305$ homes in Bexar County, Texas (TX) with four years of monthly consumption. In order to forecast the consumptions, the authors also used the building characteristics (size of living area, construction year, number of rooms, number of bedrooms, among others) and climate data (temperature and humidity). After removing some homes due to lack of data, abnormal energy consumption pattern and non-zero monthly consumptions, the final amount of homes went down to $281,779$. In the evaluation of the methods the train and test sets were mutually exclusive in terms of both households and time. Using the proposed methodology, the authors achieved a residual mean squared error (RMSE) in the prediction of the future monthly energy consumption of 99.803 ± 3.057, 100.435 ± 3.441 and 94.286 ± 3.238 kBtu/day for the linear regression, regression trees and MARS, respectively. The RMSE was measured comparing the actual average daily energy (kBtu/day) with the predicted average daily energy (kBtu/day). The authors also aggregated the daily consumption prediction to predict the monthly consumption for groups of homes. In this scenario, the authors achieved a RMSE of 25.743 ± 1.097, 18.277 ± 1.156 and 19.831 ± 1.187 kBtu/day for the linear regression, regression trees and MARS, respectively.

Rodrigues et al. [2] proposed an Artificial Neural Network (ANN) to forecast the energy consumption of households. The data used by the authors was collected from 93 households in Lisbon, Portugal. The dataset is composed by hourly energy consumption measurements for the 93 households during six to eight weeks, resulting in a total of $93,744$ records (24 hours $\times$ 7 days $\times$ 6 weeks $\times$ 93 households). The authors proposed two models, one to forecast the daily consumption and another to the hourly consumption. Both methods used an ANN with one hidden layer with 20 neurons. The input of the ANN for both methods was the energy consumption of each electric appliance (lighting, refrigerator, chest freezer, cooking, dishwasher, among others, in a total of 16 electric appliances), the apartment's area and the number of household occupants. The authors reported the ANN results in terms of Mean Absolute Percent Error (MAPE). The MAPE of the average consumption was 4.2% and the MAPE of the maximum consumption was 18.1%. Both results were reported in terms of the daily energy consumption.

Dong et al. [9] applied a hybrid approach to address the problem of forecasting the hourly and daily household energy consumption. The data used in their work was collected from four residential houses in San Antonio, TX, with different construction materials. The energy consumptions were monitored at 5-minute intervals for all the rooms. Other information was also available: outdoor temperature, solar radiation and other

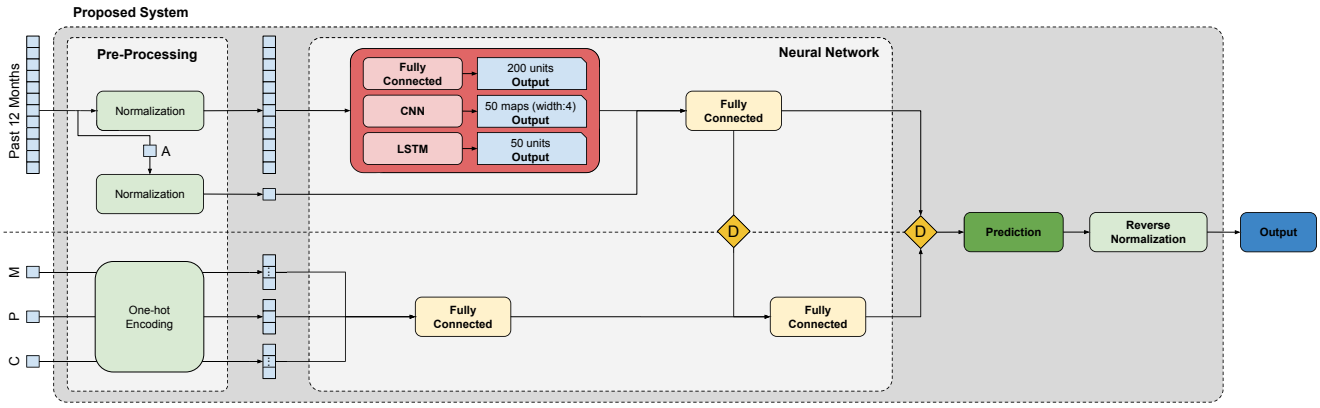


Fig. 1. System architecture. The inputs are the past 12 months, the month that is being predicted (M), the number of phases of the customer (P) and the class of the customer (C). The average of the customer consumption (A) is derived from the original input. The proposed system can be used with or without metadata (M, P and C), represented by the decision (D).

sub-metered data such as plugs, lighting, water heater and air conditioner (AC). The hybrid models proposed comprise a machine learning based step and a physic based model. The authors evaluated several machine learning methods, such as ANN, Support Vector Regression (SVR), Least-Square Support Vector Machine (LS-SVM), and others. The LS-SVM model was selected to forecast the total and non-AC energy consumptions. It was used to compose the hybrid method with a thermal network and an AC regression model to forecast the AC energy consumption: the physical model. The total energy consumption was forecasted summing up the results of the LS-SVM and the physical model. The authors trained two models, one to predict 4 hours ahead (MAPE of 15.93%) and another to predict 24 hours ahead (MAPE of 19.58%).

Although the aforementioned methods try to cope with the energy consumption forecasting problem, there are a few differences with the method proposed in this work. The method shown by Williams and Gomez [8] use more input data to forecast the energy consumption (climate data and building characteristics), and outputs the average daily consumption in the month. Rodrigues et al. [2] also use more input information. Moreover, their dataset contains the hourly consumption of each electric appliance, the apartment's area and the household occupants. The method proposed by the authors forecasts the daily consumption and the hourly consumption. In addition, the method was evaluated in only 93 households. The dataset used by Dong et al. [9] has an accurate energy consumption, measured in 5-minute intervals, information about the weather condition among others. In addition, their method was evaluated in only four households. In contrast to the presented methods, our dataset has only the monthly energy consumption and few metadata about the customers. Besides that, our dataset comprises energy consumptions of a wider range of customers (residential, industry and business), in contrast to the only residential datasets used in the literature. We try to predict the next-month energy consumption based on the 12 previous months and using a noisy data collected during electrical meters readings.

Deep neural networks have shown promising results over different research areas including but not limited to: speech recognition, image classification, video analysis and regression. Deep neural networks are architectures comprising more than one hidden layer. Examples of neural networks with deep architectures are Fully Connected neural networks, Convolutional Neural Networks, Deep Belief Networks [10], Long Short-Term Memory Networks [5], among others. In this work, the Convolutional Neural Networks and the Long Short-Term Memory Networks are investigated in the prediction of energy consumption and are better explained bellow.

The Convolutional Neural Network (CNN) was proposed by Lecun et al. [3], [4]. This network architecture was designed to work with images. Different from the common Fully Connected networks, CNNs preserve the spatial relation between features (images). In addition, this architecture was also successfully applied to non-image problems like time series analysis and speech recognition [6]. CNNs are, generally, composed by three main layers' type: convolution, subsampling and fully connected. Each input is convolved by the kernel and generates one output map. The main difference between CNNs that work with images and those that work with one-dimensional inputs is the kernel's dimensions (to the former it is a 2D or 3D kernel and to the latter it is a 1D kernel).

The Long Short-Term Memory (LSTM) is a Recurrent Neural Network (RNN) proposed by Hochreiter and Schmidhuber [5]. It was proposed to address the problem of learning to store information over extended time intervals. Different from other neural networks, LSTMs do not suffer from the vanishing gradient problem. They improve the gradient flow by the use of additive interactions. Another benefit of LSTMs (and RNNs in general) is the possibility to work on virtually unrestricted sequences, i.e. input and outputs of different sizes. LSTMs were successfully applied in many difficult problems, such as: machine translation [11], image captioning [12], automatic

speech recognition [13], and others.

III. Energy Consumption Forecast System

The system works on a temporal sequence of energy consumption, and other attributes of the customer. For each sequence, it outputs the prediction for the consumption of the next month. The general architecture of the proposed system is described in the Figure 1. We assume that, provided enough data, the consumption of one customer can be modeled by the consumption of another in the group considering 12 months as input. Therefore, the input of the system comprises the past 12 months of energy consumption of a customer, and three other attributes (referred here as input's metadata): which month is to be predicted, the number of phases of the electric power, and the class of the customer. All the non-categorical inputs are normalized. Given the normalization applied on the input data, the proposed network model outputs a normalized value. Therefore, it is required to reverse the normalization step in order to retrieve the prediction in the original data space. Finally, the prediction can be used in order to help deciding whether the value read on the electrical meter by the reader should be accepted or revised.

A. Input

The system uses multiple data (continuous and categorical) of each customer as input. One of the them is the time series of the past 12 months of energy consumption of each customer. These 12 months are sorted in a temporal sequence and given to the proposed network model after a normalization (explained later in the subsection III-B). Another customer data passed to the network model as input is the normalized average energy consumption that is derived from the past twelve months' consumption. The average is also given to the model because the normalizations applied to the past energy consumption data removes the average consumption. It can be the customer's average or the global average, depending on the normalization process. In addition to these continuous features, three categorical features are also given as input: the month that is to be predicted, e.g. January, February, December, etc.; the number of phases of the electric power and the class of the customer (a category assigned by the power company to identify similar customers, such as residential, business, industrial, etc.). Each categorical data is encoded into a one-hot representation, i.e. a N long bit-string with each bit representing a binary entry. Specifically, months are encoded into a 12-bit long one-hot vector (e.g. January is equals to 100000000000), the number of phases into a 3-bit long one-hot vector (e.g. biphasic customers are encoded into 010) and the class of the customer into a 14-bit long one-hot vector (e.g. the low-income residential customers class is encoded into 00100000000000).

B. Normalization

Two normalization techniques were used in the proposed system. Both normalizations were applied to reduce the scale and variance of the original data, expecting a more controlled behavior on the neural networks.

Standardization The first normalization is commonly used in the literature and aims to bring the whole dataset into a more confined space [14]. This space is delimited by the average and standard deviation of the training data, following Equation 1:

$$x' = \frac{x - \overline{x}}{\sigma_x} \tag{1}$$

where x and x' represent the input data and its normalized value respectively; $\overline{x}$ and σ_x are the global average and global standard deviation of the data used for training the network model. These two values are stored during training to be used for normalization and de-normalization (explained later) purposes during test. As the average consumption of each sample is also given as input to help the neural network, it is also normalized to the same space. Therefore, the average of the input ($\overline{x}_i$) is normalized based on the Equation 2:

$$\overline{x}'_i = \frac{\overline{x}_i - \overline{x}}{\sigma_x} \tag{2}$$

where $\overline{x}'_i$ is the normalized $\overline{x}_i$, $\overline{x}_i$ is the average of the original input data for the sample i.

Customer-Wise Normalization The second normalization aims to centralize the consumption data of each sample around zero and focus on the customer consumption variation. This normalization firstly subtracts the mean of each sample consumption from original consumption (Equation 3). Secondly, it computes the standard deviation of the training data after the mean subtraction, and stores this value for later use during test. Finally, it divides the mean subtracted data by the standard deviation of the mean subtracted training data (Equation 4).

$$x'_i = x_i - \overline{x_i} \tag{3}$$

$$x'_i = \frac{x'_i}{\sigma_{x'}} \tag{4}$$

where $\sigma_{x'}$ is the global standard deviation of the training data after the mean subtraction performed in Equation 3. In practical terms, this normalization exchanges the input of each customer by a variability measure of the original data around each customer's average. It introduces a new perspective of the problem to the neural network, which is to predict how much will the variability of the next month be when compared to the 12 previous energy consumptions of a given customer. Since this normalization removes the average consumption of the customers, this value is also given as a separate input to the model. To also have the customer average in a good space for the network, the average is also normalized using the standardization (Equation 2).

C. Regression

In the proposed system, an artificial neural network is used to perform the prediction of the energy consumption. This architecture can be derived into three models, by simply changing the core method (in red on the Figure 1) responsible to receive the normalized customer's energy consumption as

input. The three core methods are: a Fully Connected Neural Network (FC), a Convolution Neural Network (CNN) and a Long Short-Term Memory Network (LSTM). These three core methods, that share the rest of the architecture, are described below.

Fully Connected Neural Network (FC) On the FC-based model, the core method comprises a single hidden layer with 200 neurons. Each neuron has linear activation.

Convolution Neural Network (CNN) On the CNN-based model, the core method comprises 2 one-dimensional convolutions with a max pooling in-between. The length of the input is of 12 units. The first convolution has a kernel size of 3, with a padding of 1. This first layer produces 20 outputs with the same length of the input. Subsequently, a max pooling layer with a kernel size of 2 and stride of 2 reduces the length of the output to 6. Finally, the last convolution also has a kernel size of 3, resulting into 50 outputs, each one with length equals to 4. This was done to approximate the number of elements between the FC and CNN methods to 200.

Long Short-Term Memory Network (LSTM) On the LSTM-based model, the core method comprises a stack of 3 LSTM layers. Each layer has 256, 128 and 50 LSTM units, respectively. Each unit processes through the time dimension of the input, i.e. each LSTM unit process 12 time steps, one value at a time.

Shared Architecture The general architecture begins with the past consumption input data passing through the core method. Subsequently, the output is concatenated with the normalized average and given to a 1000-units fully connected layer. If using the variant of the model that does not use metadata, this last layer is connected to a single neuron in order to compute the final output of the model, i.e. the prediction for the next month. Otherwise, when using metadata, a parallel branch is added into the architecture. This branch concatenates all three metadata one-hot vectors into a 29 input bit-string (12 to the months, 3 to the number of phases and 14 to the class of the customer). This input is given to a 500-units fully connected layer. The length of each branch was defined to give twice the importance to the past consumption data when compared to the metadata. With this, both branch's outputs are concatenated into a 1500 vector and given to one more 1000-units fully connected layer. Finally, a single neuron is used to compute the forecast.

D. De-normalization

The network is trained with the predicted value in the normalized space. Therefore, the prediction of the network is also affected by the same normalization of the input data. In order to properly compare the predicted value with the original data, it is required to reverse the normalization after prediction. To do that, two values must have been stored when training the network: the global average ($\overline{x}$) and the global standard deviation (σ_x). The standardization of the output data can be reverted using Equation 5. The customer-wise normalization of the output requires one additional data to be stored during the training of the network: the standard deviation after the mean subtraction ($\sigma_{x'}$). With this value and the average of the customer consumption ($\overline{x_i}$), the Equation 6 can be used to reverse the normalized output data back to the original scale.

$$x = x' \times \sigma_x + \overline{x} \tag{5}$$

$$x = x' \times \sigma_{x'} + \overline{x_i} \tag{6}$$

IV. Experimental Methodology

The experiments were performed in a dataset provided by a Brazilian power company, with real energy consumption data. Three metrics, mean absolute error (MAE), mean absolute percent error (MAPE) and Median Absolute Percentage Error (MdAPE) were used to validate the results achieved. Furthermore, we follow a k-fold validation process using different configurations of the proposed normalizations and metadata. Details of these approaches are described in the next subsections.

A. Dataset

The energy consumption data come from a Brazilian power company and is collected every month by the company employees. This person goes to every customer and reads the month energy consumption from the electrical meter. Since these values are read by human employees, they are subject to error. Therefore, the data is subject to noise. In total, there were 21,978,437 measurements from 934,945 different customers. Most of the measurements (21,978,255) were obtained in a 2-year interval between March 2014 and February 2016. The measurements were grouped by customer. The samples were generated using a sliding window in time with a length of 13 out of a total of 24 months, i.e. 13 consecutive months of a customer, where the first 12 months are used in the input and the 13th is the expected value.

A filtering mechanism was used in order to select only valid samples from the measurements. These samples are subject to the following filter: the customer must have been in the electrical grid during the whole period (see invalid case of customer A in the Figure 2) and the 12-month period must contain at least one non-zero measurement (see invalid case of customer B in the Figure 2). Customers with only zero measurements are not consuming electrical power, therefore there is not enough data to forecast the energy consumption. After this filtering, 9,200,828 samples of 819,343 customers remained.

There are two main factors that distinguish samples: which month is being predicted and from which customer it was generated. The samples follow the distribution shown in the Figure 3 regarding the month of the consumption to be predicted.

Customers, in turn, are distinguished in terms of number of phases and the class assigned by the electrical company. Regarding these two factors (phase and class), the samples follow the distributions shown in the Figure 4. It can be seen in the distribution of the samples in terms of number of phases (Figure 4a) that most of samples were generated

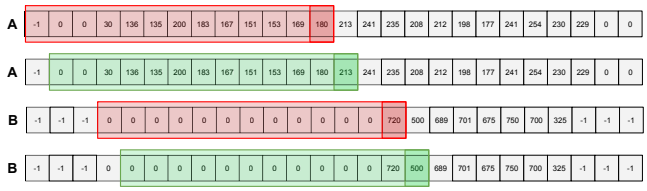


Fig. 2. Customer to Sample. A and B represents two customers. The gray boxes are monthly measurements of these customers and the colored rectangles are the sliding window used to generate samples out of customer energy consumption history. Red cases represent invalid samples, according to the constraints, and green are valid samples. The darker colored rectangles are the values to be predicted. Measurements of -1 represents that the customer was not in the electrical grid in that month.

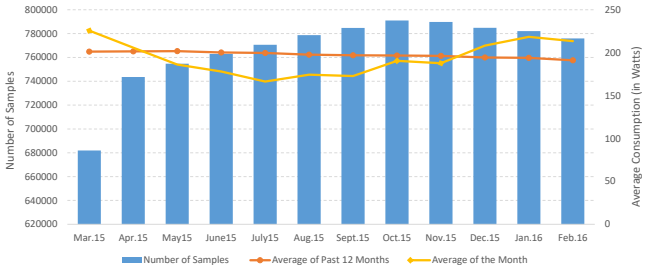
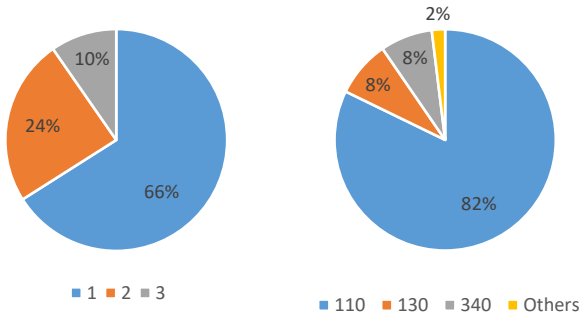


Fig. 3. Distribution of the samples across the months and the average energy consumption of the past 12 months (in orange) and the month itself (in yellow) for each of these months.



(a) Distribution of Phases (b) Distribution of Classes

Fig. 4. Distribution of the samples regarding the number of phases and their customer's class. 11 classes were grouped into "Others" because of their low representation (less than 2% of the total). The class 110 is used for residential customers, 130 for low-income residential customers and the class 340 for businesses customers.

from monophasic customers. Likewise, most of the samples came from residential customers (see Figure 4b). Also, in the Figure 4b, 11 classes were grouped into the "Others" label, because all of them represent less than 2% of the samples.

B. Metrics

In order to evaluate the proposed system, three regression metrics were used. These metrics aims to quantitatively evaluate the ability of the network to predict the energy consumption of a given customer in a month. In this way, Mean Absolute Error (MAE), Mean Absolute Percent Error (MAPE) and Median Absolute Percentage Error (MdAPE) are reported. The MAE metric is presented in Equation 7:

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}\left|\hat{f}(x_i) - y_i\right| \tag{7}$$

where n is the number of samples in the dataset, $\hat{f}(x_i)$ is the output of the system for the ith sample and y_i is the expected energy consumption, both in the original scale, i.e. after applying the reverse of the normalization.

The MAPE and MdAPE metrics weigh the error by the customer consumption (i.e a 10kWh error in a customer that consumes 40kWh is big, on the other hand it is a small error in a customer that consumes 1000kWh). Moreover, this metric was also used in other works in the literature ([2], [9]). The MAPE metric is presented in Equation 8. The variables used in these metrics are the same of the MAE metric (Equation 7).

$$\text{MAPE} = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{\hat{f}(x_i) - y_i}{y_i}\right| \tag{8}$$

The MdAPE metric is presented in Equation 9. The difference between the MAPE and MdAPE is that in the former, the mean is reported while in the latter the median is reported.

$$\text{MdAPE} = \text{median}\left\{\left|\frac{y_i - \hat{f}(x_i)}{y_i}\right|; i = 1, \ldots, m\right\} \tag{9}$$

C. Experiments

Concerning the validation of the methods, several experiments were designed. As baseline for the experiments, we replicated the procedure used by the company that provided the data as the baseline for the proposed methods. Additionally, we evaluated each configuration of the proposed models, i.e. the combinations of normalizations and metadata presence.

To perform a fair evaluation of the proposed methods, the dataset was separated using a stratified k-fold methodology. The processed dataset was split into 5 folds. For each fold, we guarantee the same distribution of samples of each combination of phase and class. As previously explained, each customer history generates multiple samples. Therefore, to allow for a fair evaluation process, we ensure that there is no overlap of customers between folds (i.e. if a customer is in the fold 0, none of its samples can be in the other folds). Each experiment uses three folds for training, one for validation and the test is carried out in the remaining fold. The validation fold was used to choose the best epoch in terms of the minimum validation loss. This procedure was used in order to prevent overfitting. The test fold was used only to evaluate the final performance of the proposed system. It is important to mention that both validation and test folds are normalized using the global average and standard deviation of the training fold. The methodology of each experiment is presented below.

Baseline The baseline for comparison follows a procedure used by the data provider, a Brazilian power company. This company assumes that the average of the past months of a customer as the predicted value. Therefore, the average of the consumption of a customer is used as baseline for comparison.

Normalization and Metadata The proposed methods can be combined regarding the normalization and the use of metadata. This combination results in four experiments: standardization with and without metadata; and customer-wise normalization with and without metadata.

Experimental Setup The implementation of the data preprocessing steps was done in-house using Python. For the FC-based and the CNN-based methods, a GPU based Deep Learning framework (Caffe [15]) was used. For the LSTM-based method, the Keras library [16] was used with TensorFlow library [17] as backend. All the experiments were carried out using an Intel Core i7 3.4 GHz with a NVIDA Tesla K40 GPU. The environment of the experiments was Linux Ubuntu 14.04, with the NVIDIA CUDA Framework 7.5 and the cuDNN library installed. The network infers at least 6,000 samples per second in the worst case, i.e. with the most computationally expensive architecture.

V. Results

As it can be seen in the Figure 5, the LSTM-based models achieved the lowest Mean Absolute Error (MAE). Moreover, the robustness of these models can be verified by the lower standard deviation when compared to the others. In the other models (CNN and FC), the customer-wise normalization considerably minimizes the MAE. In the LSTM-based models, the normalization process does not impact considerably the result given that the model itself already achieves lower averages and standard deviations. The results shown in the Figure 5 can be seen, numerically, in the Table I. The metrics (MAE, MAPE and MdAPE) and their standard deviations in the Tables I, II and III represent the average and standard deviation of the folds.

Looking closer into the results achieved by the best configuration of each model (shown in Tables II and III), the LSTM-based model indeed proved to be better than the others. As it can be seen in the Table II, the LSTM-based model achieves

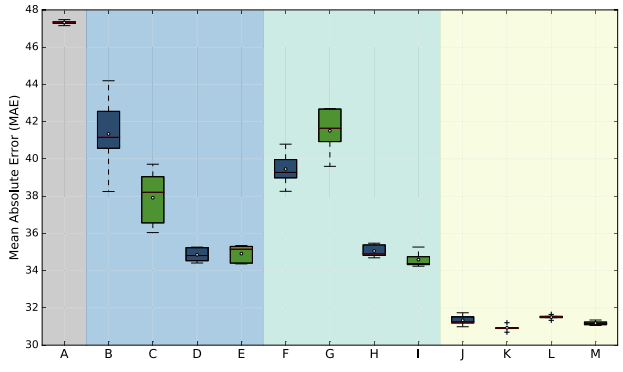


Fig. 5. The baseline model is represented by the label A in the x-axis. The FC-based models are represented by the labels B, C, D and E. The CNN-based models are represented by the labels F, G, H and I. The LSTM-based models are represented by the labels J, K, L and M. The standardization normalization was applied to the models B, C, F, G, J and K, while the customer-wise normalization to the models D, E, H, I, L and M. The metadata was used in the following models: C, E, G, I, K and M.

TABLE I
Regression Result Summary

Model	Normalization	Metadata	MAE
Baseline	—	Without	47.32 ± 0.12
FC	Standardization	Without	41.35 ± 2.23
	Standardization	With	37.92 ± 1.58
	Customer-Wise	Without	34.85 ± 0.39
	Customer-Wise	With	34.91 ± 0.49
CNN	Standardization	Without	39.45 ± 0.97
	Standardization	With	41.51 ± 1.30
	Customer-Wise	Without	35.05 ± 0.35
	Customer-Wise	With	34.59 ± 0.42
LSTM	Standardization	Without	31.34 ± 0.29
	Standardization	**With**	**30.93 ± 0.18**
	Customer-Wise	Without	31.50 ± 0.12
	Customer-Wise	With	31.18 ± 0.12

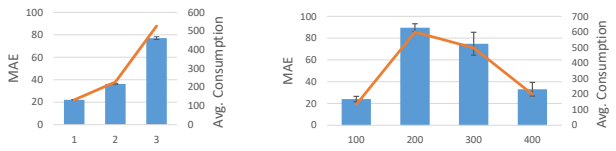
the lowest absolute error (34.6% lower than the baseline and 10.6% lower than the best of the others). In terms of relative error, the improvements are even higher (52.5% lower than the baseline and 25.7% lower than the best of the others) regarding the MAPE metric.

TABLE II
Metrics of the Best Configuration of each Model Variation

Model	MAE	MAPE	MdAPE
Baseline	47.32 ± 0.12	97.91 ± 0.92	16.71 ± 0.03
FC + CW[a]	34.85 ± 0.39	62.67 ± 2.12	12.43 ± 0.13
CNN + CW[a]+ Meta[b]	34.59 ± 0.42	62.90 ± 0.53	12.78 ± 0.30
LSTM + STD[c]+ Meta[b]	**30.93 ± 0.18**	**46.53 ± 0.28**	**10.68 ± 0.03**

[a] Customer-Wise Normalization
[b] Metadata
[c] Standardization

We have used the paired t-test [18] to carry out a pairwise statistical comparison. The results suggest that the neural network models (FC, CNN and LSTM) outperform the baseline (p-value $<$ 0.0001 for all comparisons). The LSTM-based model outperforms the CNN (p-value $=$ 3.11e$-$04) and FC (p-value $=$ 6.66e$-$05) approaches, whereas no statistical difference was found between FC and CNN based models (p-value $=$ 0.07). Further analyzing the LSTM-based model, it can be seen in the Figure 6 that the lowest absolute errors tend to happen in the lower energy consumption customer.



(a) (b)

Fig. 6. Average MAE in terms of the number of phases (Figure 6a) and the class of the customer (Figure 6b). The blue bars represent the average MAE and the orange lines depict the average consumption.

Low energy consumptions are treated differently by the power companies. There are some thresholds in which the

customers are charged equally, i.e. if the monthly consumption is less than a given threshold, the customer will be charged a minimum value. These thresholds can be seen in the Equation 10:

$$t_{min} = \begin{cases} 30 & \text{if monophasic} \\ 50 & \text{if biphasic} \\ 100 & \text{if three-phase} \end{cases} \quad (10)$$

As every consumption below these thresholds result in equal charges, they can be seen as not of interest for the problem. In total, there are 1,042,628 samples in which the value to be predicted is below these thresholds. If they are filtered out of the evaluation, the relative error of the models considerably decreases (62.84% and 7.58% lower in terms of MAPE and MdAPE, respectively), although the absolute error for the proposed system slightly increases (2.9% greater in terms of MAE), as it can be seen in the Table III. The LSTM-based model remains with the lowest absolute and relative errors, i.e. it is the best model for forecasting the monthly energy consumption in our experiments.

TABLE III
SUMMARY OF METRICS WITHOUT IRRELEVANT CONSUMPTION

Model	MAE	MAPE	MdAPE
Baseline	46.45 ± 0.13	25.54 ± 0.06	15.49 ± 0.03
FC + CW	34.98 ± 0.28	19.12 ± 0.17	11.54 ± 0.12
CNN + CW + Meta	34.72 ± 0.46	19.31 ± 0.28	11.73 ± 0.27
LSTM + STD + Meta	**31.83 ± 0.20**	**17.29 ± 0.14**	**9.87 ± 0.03**

In addition to these deep-learning-based models, other well-known regressors were evaluated: linear, ridge, Partial Least Squares (PLS) and Stochastic Gradient Descent (SGD) regressors. The best result of the regressors was achieved by the SGD with the standardization: 36.54 and 14.4 of MAE and MdAPE, respectively. Despite of being better than the baseline, the results of the regressors were worse than the FC-based result.

Most of the data used in this research area is given by private companies. As a result, none of the most relevant works share their datasets. This hinders the comparison of results across the literature. This can be confirmed by the fact the none of the works referenced in our literature review compared their results with others. In our case, we were able to compare with the baseline method used by the power company, and it shows a considerable improvement in the method.

VI. CONCLUSION

We have presented a deep-learning based model for the problem of monthly energy consumption forecast, a time series regression task. Moreover, we have provided an analysis of these models in a large dataset with almost 10 million samples from almost one million customers. Results showed the proposed system outperforms the baseline method that is currently used by some power companies. Additionally, three architectures were evaluated: fully connected (FC), Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) Networks. Results showed the LSTM-based model is the best for the proposed problem in terms of absolute and relative errors. The proposed system can process at least 6,000 samples per second. Ultimately, it confirms that LSTMs are more suitable for temporal task analysis such as the one hereby tackled.

ACKNOWLEDGMENT

This work was partially supported by EDP Escelsa and EDP Bandeirante. We also would like to thank UFES for the support and CAPES for the scholarships. We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Tesla K40 GPU used for this research.

REFERENCES

[1] International Energy Agency, "Key world energy statistics," 2016, accessed 11 November 2016. [Online]. Available: https://www.iea.org/publications/freepublications/publication/KeyWorld2016.pdf
[2] F. Rodrigues, C. Cardeira, and J. Calado, "The daily and hourly energy consumption and load forecasting using artificial neural network method: A case study using a set of 93 households in portugal," *Energy Procedia*, vol. 62, pp. 220 – 229, 2014.
[3] Y. LeCun, B. E. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. E. Hubbard, and L. D. Jackel, "Handwritten digit recognition with a back-propagation network," in *Advances in Neural Information Processing Systems 2*, 1990, pp. 396–404.
[4] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.
[5] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
[6] Y. LeCun and Y. Bengio, "Convolutional networks for images, speech, and time series," *The handbook of brain theory and neural networks*, vol. 3361, no. 10, p. 1995, 1995.
[7] H. xiang Zhao and F. Magoulès, "A review on the prediction of building energy consumption," *Renewable and Sustainable Energy Reviews*, vol. 16, no. 6, pp. 3586 – 3592, 2012.
[8] K. T. Williams and J. D. Gomez, "Predicting future monthly residential energy consumption using building characteristics and climate data: A statistical learning approach," *Energy and Buildings*, vol. 128, pp. 1 – 11, 2016.
[9] B. Dong, Z. Li, S. M. Rahman, and R. Vega, "A hybrid model approach for forecasting future residential electricity consumption," *Energy and Buildings*, vol. 117, pp. 341 – 351, 2016.
[10] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural Comput.*, vol. 18, pp. 1527–1554, 2006.
[11] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Advances in neural information processing systems*, 2014, pp. 3104–3112.
[12] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan, "Show and tell: A neural image caption generator," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3156–3164.
[13] J. Li, A. Mohamed, G. Zweig, and Y. Gong, "Lstm time and frequency recurrence for automatic speech recognition," in *2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*. IEEE, 2015, pp. 187–191.
[14] E. Kreyszig, *Advanced Engineering Mathematics*. John Wiley & Sons Inc, 1979.
[15] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," *arXiv preprint arXiv:1408.5093*, 2014.
[16] F. Chollet, "Keras," https://github.com/fchollet/keras, 2015.
[17] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, M. Kudlur, J. Levenberg, R. Monga, S. Moore, D. G. Murray, B. Steiner, P. Tucker, V. Vasudevan, P. Warden, M. Wicke, Y. Yu, and X. Zheng, "Tensorflow: A system for large-scale machine learning," in *12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16)*, GA, 2016, pp. 265–283.
[18] G. Casella and R. L. Berger, *Statistical Inference*. Duxbury Pacific Grove, CA, 2002, vol. 2.