# An Expanded Study of the Application of Deep Learning Models in Energy Consumption Prediction

Leonardo Santos Amaral[2](✉), Gustavo Medeiros de Araújo[2], Ricardo Moraes[2], and Paula Monteiro de Oliveira Villela[1]

[1] Universidade Estadual de Montes Claros - UNIMONTES, Av. Prof. Rui Braga, S/N - Vila Mauriceia, Montes Claros – MG, Brazil
[2] Universidade Federal de Santa Catarina UFSC, R. Eng. Agronômico Andrei Cristian Ferreira, S/N -Trindade, Florianópolis – SC, Brazil
leonardosamaral@hotmail.com, {gustavo.araujo, ricardo.moraes}@ufsc.br

**Abstract.** The time series of electrical loads are complex, influenced by multiple variables (endogenous and exogenous), display non-linear behavior and have multiple seasonality with daily, weekly and annual cycles. This paper addresses the main aspects of demand forecast modeling from time series and applies machine learning techniques for this type of problem. The results indicate that through an amplified model including the selection of variables, seasonality representation technique selection, appropriate choice of model for database (deep or shallow) and its calibration, it's possible to archive better results with an acceptable computational cost. In the conclusion, suggestions for the continuity of the study are presented.

**Keywords:** Forecast · Energy · Demand · Deep learning

## 1 Introduction

Global energy consumption is growing each year, and to satisfy this demand, it is necessary to find ways to make efficient use of this resource and generate more energy. In 2010, global energy consumption was approximately 152 thousand TWh, and this value increased to 173 thousand TWh in 2017, which represents a 13% growth in seven years [10].

At present, the sector is in a transition phase and some changes are happening, such as: increase of the use of renewable sources, increased energy efficiency, modernization of the network infrastructure, consumption patterns changes and deregulation [5]. There is currently greater awareness of the environmental impacts caused by high energy consumption, and despite this increasing consumption, the challenge is to reduce energy production from polluted energy sources such as nuclear and fossil fuels.


A. L. Pinto and R. Arencibia-Jorge (Eds.): DIONE 2022, LNICST 452, pp. 150–162, 2022.
https://doi.org/10.1007/978-3-031-22324-2_12

On a global level, it is essential that the decision makers have information on the energy consumption growth in different countries to meet these challenges. On a local level, this information is also essential due to changes in the sector, such as greater competitiveness, measures to rationalize consumption, and economic interests. The operation of a Smart-Grid network dependent on this information is one example, as it is sensitive to instantaneous load variations, contingencies, and the economic dispatch of loads.

Some energy forecasting models are based purely on statistical methods; however, in the last decades there has been an increase in the use of methods based on machine learning (ML), which make use of the large amount of information currently available to make predictions and obtain more accurate results. Numerous works published in recent decades have addressed demand forecasting, but most do not consider all relevant aspects of it. This work aims to fill this gap by indicating ways to build more complete models.

## 2 Related Works

Over the last few decades, several demand forecasting techniques have been tested using different strategies to obtain future values of energy consumption. For example, [7] classifies different methods in terms of use; multiple regression or multivariate regression methods are the most used, followed by the ANN (artificial neural network). The most used time series models include ARMA (autoregressive moving average) for stationary series and ARIMA (autoregressive integrated moving average) if the series is non-stationary and requires differentiation. The next in the author's ranking is the SVM (support vector machine) and finally, there are models with small participation at the same level.

In a survey, [11] points out that some models offer advantages over others if the objective is forecasting demand. Multiple regressions provide accuracy in the prediction when the load in time follows a linear behavior. Neural networks are preferred when the time variations are significantly fast. ANN and time series models are often used for STLF (short term load forecasting) when the consumption pattern is more complex. Time series models rely mainly on historical load data, while ANN needs to be trained with the historical load data and the time base. SVM is very similar to ANN, as it also requires parameters as input, but it is faster and has fewer parameters to configure. Evolutionary algorithms run faster and provide better results when used with ANN. Over time, some methods were shown to be ineffective for short-term forecasting demand due to the non-linear relationship between the predicted variables and the predictors, along with the increase in the amount of information.

ML techniques have delivered good results for this type of problem. For example, [8] highlights that, during the last two decades, several ML models for forecasting demands have emerged, because ML offers the best opportunities to search for improvements. Some advantages include the increase in accuracy and robustness, as well as the ability to generalize when creating hybrid models. Among ML techniques, the most promising involve deep learning. In [9], a comparison was made between traditional ML techniques and deep-learning techniques, concluding that the latter leads to better results, although its model preparation presents bigger challenges.

# 3 Problem and Motivation

When making demand forecasts, some questions arise:

- Which elements of influence should be considered and how? (i.e., seasonality and input variables)
- How should the most relevant variables be selected and the redundant ones eliminated for better computational performance and results?
- Which prediction technique should be used?
- How should the model be calibrated?

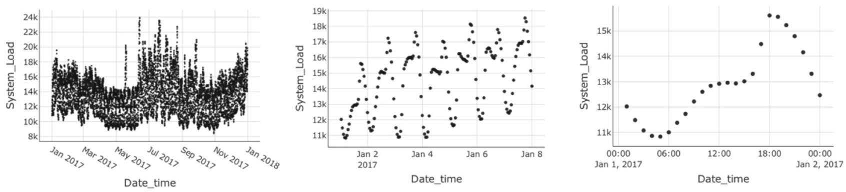
There are few publications that attempt to answer all these questions simultaneously, and this fact motivated the development of this study, whose purpose is to build an expanded model that contemplates all these aspects simultaneously.

# 4 Proposed Model

## 4.1 Database

The database used in this case study is available at ISO NE (Independent System Operator New England) at http://www.iso-ne.com. It includes the sum of the global electrical load data from several cities in England for the period between January 2017 and December 2019. Altogether, it contains 23 independent variables of the following natures: climate information, economic indicators, and market data.

To better understand demand behavior, Figs. 1, 2, and 3 show its annual, weekly, and daily variation. The concentration of peaks throughout the year appear between the months of June and September, over the week on Mondays, and by day between 6 p.m. and 7 p.m. This behavior reflects the presence of multiple seasonalities.

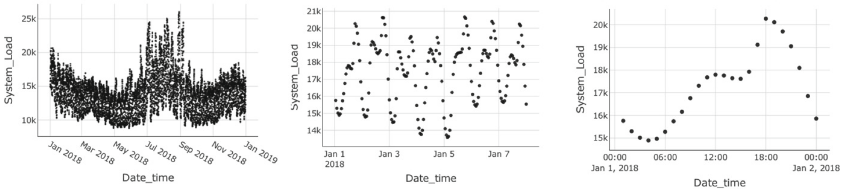



a)Annual behavior b)Weekly behavior c)Daily behavior

**Fig. 1.** Variation in energy consumption throughout 2017 (Source: the author)

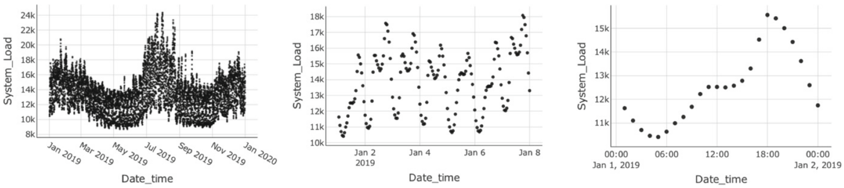
It is very important that the database includes the variables with greatest influence on energy consumption when making predictions. For example, the graphs in Figs. 4, 5, and 6 reproduce the behavior of temperature throughout the day. The temperature variations are shown in the same periods: throughout the year, week, and day.

As shown in Figs. 4, 5, and 6, the temperature peaks from 2017 to 2019 occurred between the months June and October; weekly, they often occur on Wednesdays; and daily, between 5 p.m. and 6 p.m. These data indicate the existence of a correlation between temperature and the target variable, energy consumption.



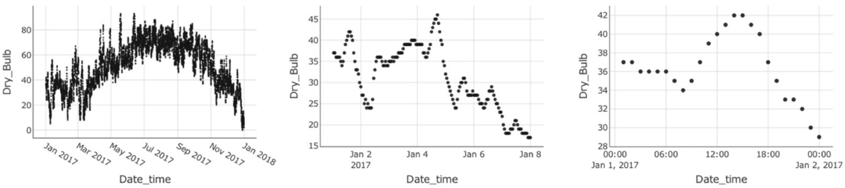

a)Annual behavior b)Weekly behavior c)Daily behavior

**Fig. 2.** Variation in energy consumption throughout 2018 (Source: the author)



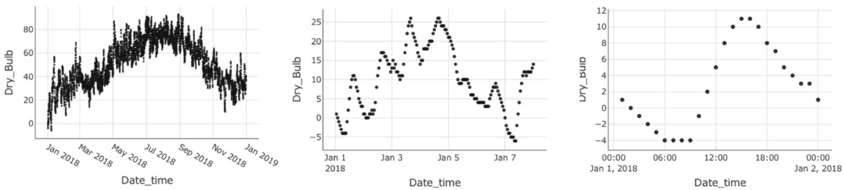

a)Annual behavior b)Weekly behavior c)Daily behavior

**Fig. 3.** Variation in energy consumption throughout 2019. (Source: the author)



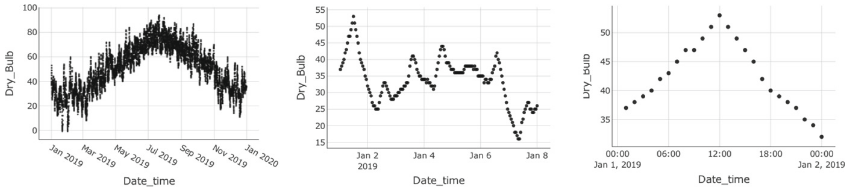

a)Annual behavior b)Weekly behavior c)Daily behavior

**Fig. 4.** Dry bulb temperature variation in 2017 (Source: the author)




a)Annual behavior b)Weekly behavior c)Daily behavior

**Fig. 5.** Dry bulb temperature variation in 2018 (Source: the author)




a)Annual behavior b)Weekly behavior c)Daily behavior

**Fig. 6.** Dry bulb temperature variation in 2019 (Source: the author)

## 4.2 Load Seasonality

As noted in Sect. 4.1, the electrical load has multiple seasonalities. This issue is addressed in [3], which proposes different ways of coding to represent them. They are shown below.

**Representation of the Year Period**

To encode the variations caused by hot and cold periods throughout the year, a trigonometric function is used that generates a specific value for each day of the year. The expression is shown below:

$$p = [\sin(2\pi \frac{(i+\tau)}{366}) \ \cos(2\pi \frac{(i+\tau)}{366})]. \tag{1}$$

where i represents the current day and $\tau$ the prediction range in days. So $i + \tau = 1$, 2,…366 represents the number of the day of the year of the forecast demand.

**Representation of the Day of the Week**

The day of the week varies between 1 (Monday) and 7 (Sunday), and six different encodings are used to represent them, listed in Table 1. One scales the day of the week between [1 … 1/7]: in this successive encoding, days of the week have successive scaled index values. A trigonometric expression is used to encode the index of the day in the second representation, and the following ones use bits to represent each day of the week.

**Table 1.** Representation of the day of the week

| d | 1<br>$d^1$ | 2<br>$d^2$ | 3<br>$d^3$ | 4<br>$d^4$ | 5<br>$d^5$ | 6<br>$d^6$ |
|---|---|---|---|---|---|---|
| Monday | δ/7 | $[\sin(2\pi \frac{(\delta)}{7}) \ \cos(2\pi \frac{(\delta)}{7})]$ | 1000000 | 001 | 001 | 00 |
| Tuesday | | | 0100000 | 010 | 011 | 01 |
| Wednesday | | | 0010000 | 011 | 010 | 01 |
| Thursday | | | 0001000 | 100 | 110 | 01 |
| Friday | | | 0000100 | 101 | 111 | 01 |

(*continued*)

**Table 1.** (*continued*)

| d | 1<br>$d^1$ | 2<br>$d^2$ | 3<br>$d^3$ | 4<br>$d^4$ | 5<br>$d^5$ | 6<br>$d^6$ |
|---|---|---|---|---|---|---|
| Saturday | | | 0000010 | 110 | 101 | 10 |
| Sunday | | | 0000001 | 111 | 100 | 11 |

Source: [3]

**Representation of the Hour of the Day**

The representation of the time of the day is made by five encodings, as shown in Table 2. The first scale is the time of the day [1 … 24], successive hours of the day having successive scaled index values. The second representation has a trigonometric expression that generates values for each hour of the day, and for representations 3, 4, and 5 bits are used to represent the hour of the day.

**Table 2.** Representation of the hour of the day

| h | 1<br>$h^1$ | 2<br>$h^2$ | 3<br>$h^3$ | 4<br>$h^4$ | 5<br>$h^5$ |
|---|---|---|---|---|---|
| Hour 1 | δ/24 | $[\sin(2\pi t\frac{(\delta)}{24})\ \cos(2\pi t\frac{(\delta)}{24})]$ | 1000…0 | 00001 | 00001 |
| Hour 2 | | | 0100…0 | 00010 | 00011 |
| Hour 3 | | | 0010…0 | 00011 | 00010 |
| … | | | … | … | … |
| Hour 24 | | | 0000…1 | 11000 | 10100 |

Source: [3]

### 4.3 Consumption Prediction from Time Series

It is possible to make future prediction of H energy values from a univariational time series comprised of T observations [12]. Typically, energy data are characterized by high variance and non-linear behavior between independent and dependent variables.

As pointed out in the cited works in Sect. 2, due to these characteristics, deep-learning techniques have stood out in demand forecastting applications based on time series. For this reason, this work focuses on using deep-learning models to predict energy consumption. According to [1], several steps are necessary to assemble a good model, as shown in Fig. 7.

The preprocessing step is very important, because helps to reduce the database dimensionality while keeping the most relevant information [2]. According to [2], some of the offered advantages are reduced processing time, increased accuracy, and the possibility of a better understanding of the learning model and data. He classifies the variable

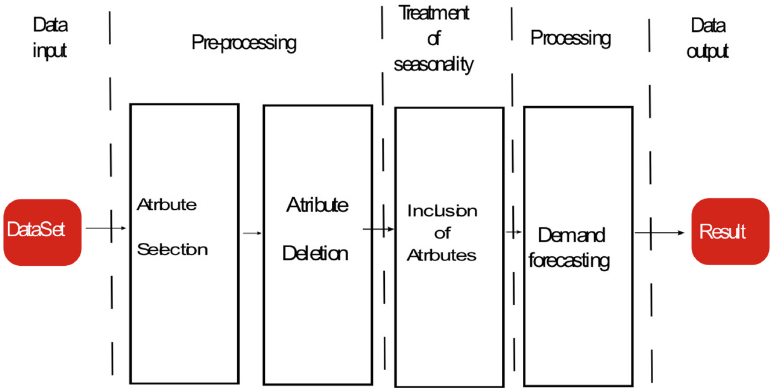



**Fig. 7.** Framework for energy prediction (Source: Author)

selection methods as: filter, wrapper, and embedded. The next step is the seasonality treatment, which consists of adding attributes to reproduce it.

Due to the advantages exposed in Sect. 2, it was decided to make the predictions using SVM, a classic and widely used ML method, and two deep-learning techniques: the multi layer Perceptron and LSTM (long short-term memory). It will thus be possible to evaluate the potential of the proposed models.

# 5 Implementation and Results

## 5.1 Variable Selection and Model Entry

The increase in the database dimensions with a large supply of data exposes the limitations of ML algorithms in terms of learning capacity and processing time. A high number of input variables makes the processing heavy without necessarily delivering better results [6]. For this reason, a preprocessing step is included to remove irrelevant and redundant data. In this step, the whole database is submitted to algorithms capable of identifying the most relevant attributes; then, the result of this selection is submitted to the RFM (recursive feature elimination) algorithm to eliminate redundant attributes.

The database was loaded into the WEKA tool and processed through attribute-selection algorithms. Weka is a free ML tool developed in Java at the University of Waikato, which has the main attribute-selection algorithms implemented and offers easy access. More information about WEKA's variable selection algorithms can be found in its manual.

The algorithms were used in Table 3 to allow a relevance and redundancy analysis on the database attributes. CFS privileges subsets that have less intercorrelation with each other but are highly correlated with the target [4]. The ClassifierAttributeEval checks the value of an attribute used with a specific classifier selected by the user. The main component is used to extract more relevant information from a redundant and noisy dataset. Relief is a well-known method that measures the relevance of attributes close to the target. Mutual information focuses on relevance analysis using the concept of mutual information [6].

Below, Table 3 identifies the selection method and the attributes considered most relevant. The purpose is to reduce the dimensionality of the database, so the main attributes were selected. The results of the pre-processing are shown in Table 3, which identifies the algorithm and the most relevant attributes.

**Table 3.** Attribute selection

| Algorithm | Selection criteria | Stop criterion | Result | Used for | Problem type | Source |
|---|---|---|---|---|---|---|
| CfsSubsetEvail | Subset | **Greedy** Stepwise | RT demand | Eliminate attributes | Nonlinear | WEKA |
| | | | DA_CC | | | |
| | | | DA MLC | | | |
| | | | RT MLC | | | |
| | | | Min_5min_RSP | | | |
| | | | Max_5min_RSP | | | |
| ClassifierAtributeEvail | Individual | Ranking | Max_5min_RCP | Eliminate attributes | Nonlinear | WEKA |
| | | | DA_EC | | | |
| | | | DA_CC | | | |
| | | | RT demand | | | |
| | | | DA LMP | | | |
| PrincipalComponentes | Individual | Ranking | RT LMP | Eliminate attributes | Nonlinear | WEKA |
| | | | RT_EC | | | |
| | | | DA LMP | | | |
| | | | DA_EC | | | |
| | | | RT MLC | | | |
| Relief | Subset | Ranking | RT demand | Eliminate attributes | Nonlinear | WEKA |
| | | | DA demand | | | |
| | | | DA_EC | | | |
| | | | DA_LMP | | | |
| | | | Reg_Service_Price | | | |
| Mutual information | Individual | Ranking | DA_ MLC | Eliminate attributes | Nonlinear | Python Code |
| | | | DA_LMP | | | |
| | | | Min_5 min_RSP | | | |
| | | | DA_EC | | | |
| | | | Dew Point | | | |

Source: Author

It is important to highlight that this step only considered the original attributes of the database, so it did not consider the attributes that will be added due to the reproduction of seasonality of the electrical load.

## 5.2 Comparison of the Models Results

To test three different energy consumption prediction models, several simulations were performed using the database described in Sect. 4.1. The first model tested used the SVM as the estimator, the second used the multilayer Perceptron, and the third uses the LSTM. As discussed above, these estimators were chosen because they are effective in making estimates from time series. The model using SVM is considered a traditional ML model and was calibrated to solve the problem and be a reference in the comparison with deep neural network models. In Figs. 8 and 9 below, two model variants, one shallow and the other deep, will be considered in the analysis. What distinguishes the two is the number of intermediate layers. Note that shallow models have only one input layer, followed by a dense layer that generates the output values. The deep model has different layers of the relevant class in series and an output dimensionality similar to the shallow model.

| | | |
|---|---|---|
| Input Layer: Perceptron | Input | 456 |
| | Output | 8 |
| Output layer: Perceptron | Input | 8 |
| | Output | 24 |

| | | |
|---|---|---|
| Input layer: LSTM | Input | (none,1,456) |
| | Output | (none,1,50) |
| Output layer: LSTM | Input | (none,1,50) |
| | Output | (none,1,50) |

**Fig. 8.** Shallow models (Source: Author)

| | | |
|---|---|---|
| Input Layer: Perceptron | Input | 456 |
| | Output | 8 |
| Middle Layer: Dense | Input | 8 |
| | Output | 8 |
| Middle Layer: Dense | Input | 8 |
| | Output | 8 |
| Middle Layer: Dense | Input | 8 |
| | Output | 8 |
| Output layer: Dense | Input | 8 |
| | Output | 24 |

| | | |
|---|---|---|
| Input Layer: LSTM | Input | (none,1,456) |
| | Output | (none,1,50) |
| Middle Layer: LSTM | Input | (none,1,50) |
| | Output | (none,1,50) |
| Middle Layer: LSTM | Input | (none,1,50) |
| | Output | (none,1,50) |
| Middle Layer: LSTM | Input | (none,1,50) |
| | Output | (none,1,50) |
| Output layer: Dense | Input | (none,1,50) |
| | Output | (none,1,1) |

**Fig. 9.** Deep models (Source: Author)

In this study, simulations were performed to calibrate the models, and the results are shown in Tables 4, 5 and 6 below. They indicate the values of the main parameters of the models.

The simulations in Table 4 indicate that increasing the value of C and using the RBF kernel increases the accuracy of the model. However, it should be noted that in this condition greater computational effort is required to process the data. For example, when migrating from C = 20 to C = 2000, the processing time increases from 0.6 to 14.4 min.

**Table 4.** SVM Model processes

| C | Epsilon | Rbf | | Polynomial | |
|---|---|---|---|---|---|
| | | MAPE | RMSE | MAPE | RMSE |
| 20 | 0,01 | 2,09 | 869 | 9,24 | 1553 |
| | 0,1 | 2,09 | 869 | 9,24 | 1553 |
| 200 | 0,01 | 1,0 | 590 | 7,37 | 1598 |
| | 0,1 | 1,0 | 590 | 7,37 | 1598 |
| 2000 | 0,01 | 0,53 | 374 | 7,04 | 1888 |
| | 0,1 | 0,53 | 374 | 7,03 | 1887 |

Source: Author

**Table 5.** Perceptron model processes

| epoch | Optimizer | Activation Function | Test (%) | Error | |
|---|---|---|---|---|---|
| | | | | MAPE | RMSE |
| 50 | RMSProp | relu/tanh | 20 | 3,54 | 617 |
| | RMSProp | relu | 20 | 5,6 | 932 |
| | RMSProp | relu/tanh | 10 | 2,85 | 607 |
| | RMSProp | relu | 10 | 4,44 | 760 |
| | Adam | relu/tanh | 10 | 2,29 | 500 |
| | Adam | relu | 10 | 4,26 | 811 |
| 100 | RMSProp | relu/tanh | 20 | 3,22 | 741 |
| | RMSProp | relu | 20 | 4,78 | 1006 |
| | RMSProp | relu/tanh | 10 | 2,71 | 573 |
| | RMSProp | relu | 10 | 5,03 | 990 |
| | Adam | relu/tanh | 10 | 2,72 | 619 |
| | Adam | relu | 10 | 4,28 | 838 |

Source: Author

Table 5 shows the simulations performed with the model using Perceptron, with the main Perceptron hyperparameters.

From the results of Table 5, it can be observed that both the RMSprop and Adam optimizers offer satisfactory performance, and that the use of the relu activation function in the intermediate layers and tanh in the output layers also contributed to delivering better results. It is also evident that by increasing the training percentage and the number of epochs, the accuracy of the model improves. Similar to the other cases, Table 6 includes processing performed by varying the main parameters of the LSTM.

**Table 6.** LSTM model processes

| Epoch | Optimizer | Activation function | Loss | Error | |
|---|---|---|---|---|---|
| | | | | MAPE | RMSE |
| 50 | RMSProp | relu | mse | 3,95 | 519 |
| | RMSProp | relu | mae | 3,85 | 512 |
| | RMSProp | tanh | mse | 4,18 | 549 |
| | RMSProp | tanh | mae | 3,84 | 511 |
| | RMSProp | sigmoid | mse | 4,72 | 613 |
| | Adam | sigmoid | mse | 3,85 | 522 |
| 100 | RMSProp | relu | mse | 4,10 | 560 |
| | RMSProp | relu | mae | 4,19 | 583 |
| | RMSProp | tanh | mse | 3,97 | 520 |
| | RMSProp | tanh | mae | 3,78 | 504 |
| | Adam | relu | mse | 3,95 | 540 |

Source: Author

Once again, it was possible to observe that both the Adam and RMSprop optimizers offer good performance and that the use of the tanh activation function also contributes to achieving better results. Error functions do not significantly influence the results.

The next step after calibrating the models is to evaluate their response to different input conditions. Table 7 includes seven different input situations and the respective responses of the models. The first considers all the attributes of the base and disregards seasonality, and the following three consider only the main variables selected by some of the methods indicated in Table 3, also disregarding seasonality. The last three considered the same entries as the previous items with the addition of seasonality variables.

The results in Table 7 indicate that the highest model accuracy was achieved when all independent variables were considered in their modeling (first line of Table 7). The accuracy achieved with the SVR-based model exceeded that obtained with the deep-learning models. This result demonstrates that it is a simpler method to use and may be more useful depending on the database size. When reducing the dimensionality of the database, the accuracy of the models was reduced, although the benefit in terms of computational cost justifies doing so. Regarding the impact of the selection of variables, the best results were obtained in most processing using the CFS.

In some simulations, improvements were observed resulting from the implementation of seasonality, however, the need to test other forms of coding was evident. In this work, only codes using trigonometric expressions were tested. The results indicate that, for this particular database, the shallow models generated better results than deep ones.

**Table 7.** Simulations with demand prediction models

| Selection Attributes (criteria) | Seasonality | SVR | | PERCEPTRON | | | | LSTM | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | Shallow | | Deep | | Shallow | | Deep | |
| | | mape | rmse | mape | rmse | mape | rmse | mape | rmse | mape | rmse |
| 100% da base | n | 2,09 | 869 | 3,26 | 550 | 3,36 | 615 | 3,76 | 633 | 3,97 | 650 |
| CFS | n | 1,63 | 777 | 5,41 | 1017 | 6,09 | 1247 | 3,83 | 514 | 3,94 | 671 |
| Relief | n | 1,15 | 659 | 5,47 | 1116 | 5,60 | 1100 | 3,94 | 669 | 4,17 | 727 |
| Mutual information | n | 9,58 | 166 | 5,43 | 1091 | 5,81 | 1154 | 3,88 | 652 | 4,03 | 690 |
| CFS | s | 2,22 | 825 | 5,23 | 1056 | 5,66 | 1150 | 3,84 | 654 | 4,10 | 718 |
| Relief | s | 1,59 | 728 | 5,23 | 10,37 | 5,4 | 1054 | 3,86 | 653 | 4,01 | 696 |
| Mutual information | s | 9,74 | 1674 | 4,79 | 965 | 5,78 | 1192 | 3,86 | 517 | 4,12 | 715 |

Source: Author

## 6 Conclusion

The main objective of this study was to show the potential of ML techniques to forecast energy demand from time series and illustrate the main challenges in building more complete models. As explained in Sect. 4.3, to overcome the limitations of ML techniques, it is recommended to reduce the database dimensionality, selecting the input variables for the model. The results indicate that, with a reduced number of variables, it is possible to build a model with an accuracy very close to that achieved with a model that uses all the attributes of the database, as shown by the results achieved with the CFS method using the SVR, which generated a map of 2.09 against 1.15, and with LSTM, which generated a map of 3.76 against 3.83.

When comparing the simulations with and without the representation of seasonality, it is possible to observe the improvement in the model response in parts of the simulations. This shows the need to test other forms of coding to expand the benefits of representing this phenomenon in the model. The results also showed that, for the database used, shallow models were more appropriate for modeling, as they required less computational effort and presented better results. However, as the database grows, it is likely that this behavior will change and the accuracy of deep models will exceed the accuracy of shallow ones.

Future work could consider tests with larger databases, using a greater number of exogenous variables (irradiation level, economic indicators and others), new ways of representing seasonality and reflections of electric energy generations distributed.

## References

1. Ahmad, W., et al.: Towards short term electricity load forecasting using improved support vector machine and extreme learning machine. Energies **13**, 2907 (2020)

2. Cai, J., Luo, J., Wang, S., Yang, S.: Feature selection in machine learning: a new perspective. Neurocomputing **300**, 70–79 (2018)
3. Dudek, G.: 'Multilayer perceptron for short-term load forecasting: from global to local approach. Neural Comput. Appl. **32**(8), 3695–3707 (2020)
4. Gnanambal, S., Thangaraj, M., Meenatchi, V., Gayathri, V.: Classification algorithms with attribute selection: an evaluation study using WEKA. Int. J. Adv. Netw. Appl. **9**, 3640–3644 (2018)
5. Keitsch, K.A., Bruckner, T.: Input data analysis for optimized short term load forecasts. In: IEEE Innovative Smart Grid Technologies, Asia (ISGT-Asia), Melbourne, VIC, pp. 1–6 (2016)
6. Khalid, S., Khalil, T., Nasreen, S.: A survey of feature selection and feature extraction techniques in machine learning. In: 2014 Science and Information Conference, pp. 372–378 (2014)
7. Kuster, C., Rezgui, Y., Mourshed, M.: Electrical load forecasting models: a critical systematic review. Sustain. Cities Soc. **35**, 257–270 (2017)
8. Mosavi, A., Salimi, M., Ardabili, S.F., Rabczuk, T., Shamshirband, S., Várkonyi-Kóczy, A.R.: State of the art of machine learning models in energy systems, a systematic review. Energies **12**, 1301 (2019)
9. Paterakis, N.G., Mocanu, E., Gibescu, M., Stappers, B., Van Alst, W.: Deep learning versus traditional machine learning methods for aggregated energy demand prediction. In: IEEE PES Innovative Smart Grid Technologies Conference Europe, ISGT-Europe – Proceedings (2017)
10. Smil, V.: Energy Transitions: Global and National Perspectives. ABC-CLIO, , Santa Barbara (2017)
11. Upadhaya, D., Thakur, R., Singh, N.K.: A systematic review on the methods of short-term load forecasting. In: Proceedings of the 2019 2nd International Conference on Power Energy, Environment and Intelligent Control (PEEIC), Greater Noida, India, 18–19 October, pp. 6–11 (2019)
12. Müller, K.-R., Smola, A.J., Rätsch, G., Schölkopf, B., Kohlmorgen, J., Vapnik, V.: Predicting time series with support vector machines. In: Gerstner, W., Germond, A., Hasler, M., Nicoud, J.-D. (eds.) ICANN 1997. LNCS, vol. 1327, pp. 999–1004. Springer, Heidelberg (1997). https://doi.org/10.1007/BFb0020283