Hyperparameter Optimization of Deep Learning model for Short-term Electricity Demand Forecasting

Félix Morales, Carlos Sauer, Hans Mersch, Diego Stalder
Facultad de Ingeniería
Universidad Nacional de Asunción
Asunción, Paraguay
felixmorales@fiuna.edu.py, csauer@fiuna.edu.py, hmersch@fiuna.edu.py, dstalder@ing.una.py

Miguel García Torres
Data Science and Big Data Lab
Universidad Pablo de Olavide
España, Sevilla
mgarciat@upo.es

Abstract

Short-term electricity demand forecasting represents a fundamental tool for decision-making by entities engaged in electricity management since it allows the development of strategies to meet variations in electricity demand in short periods. The accuracy of predictive models is an important factor for energy operations and the scheduling of energy generation sources to meet the demand at each instant. Intelligent models based on Recurrent Neural Networks (RNN) require hyperparameter adjustment. These models have several hyperparameters that substantially affect their performance. Our paper implements a Long-Short Term Memory (LSTM) model and four search methods to adjust its hyperparameters. First, we select the length of historical window and the hidden state size of LSTM cells for optimization. Second, we draw comparisons between the grid search, random search, a Bayesian scheme, and a genetic algorithm. The data set used for training and validation of the model includes hourly electricity consumption and meteorological variables recorded in Paraguay from 2015 to 2021. The proposed model was evaluated through numerical experiments with classical error measures such as the root mean square error (RMSE), the correlation, the runtime, and the mean absolute percentage error (MAPE). Our comparative study shows that grid search and genetic algorithm give the optimal hyperparameters with high validation accuracy on the test dataset. However, it is important to note that grid search may require much more evaluations and computational resources.

Keywords
Recurrent Neural Networks, Electric Demand, Forecasting, Optimization, Genetic Algorithms

1. Introduction
Electricity demand forecasting is a widely studied topic of great interest for electric companies in any country. It is an important element for making strategic decisions, such as the scheduling of generating units and the planning of operations to meet the demand of the electric system. If the demand forecast is very different from the one that finally occurs, the need for additional contingency generation or the flexibility to reduce generation can increase costs. Electric companies plan on different time scales (annual, monthly, weekly, daily, and hourly). Although the demand is separated into components: trend, cyclical or periodic movements, seasonal variation, and finally irregular movement, it is still difficult to make accurate forecasts in the short term (hourly) (Seo, 2012) . This is because electricity demand depends on consumer behavior and the same can be affected by weather factors or daily routines.

Forecasting methods based on machine learning are commonly used for this type of scenario (Gallardo, 2021). When the problem is centered on the provision of several units, it is normal to perform a clustering first to reduce the models to be used (Morales, 2022). However, models based on deep learning have shown better performance for more complex problems (Bouktif, 2018) (Bogado, Stalder, Gómez, & Schaerer, 2020) (Bogado, Stalder, Schaerer, & Gómez-Guerrero, 2021). These algorithms have several hyperparameters whose optimal settings determine even better results. Nevertheless, this adjustment presents a laborious task due to the number of existing combinations. To deal with this, meta-heuristic genetic algorithms can be applied to select parameters such as the number of LSTM units, number of layers, and neurons in each layer, among other parameters (Cheng, 2017). We use the DEAP (Distributed Evolutionary Algorithms in Python) library to implement the genetic algorithm (Gardner, 2012).

1.1 Objectives
Implement hyperparameter optimization algorithms such as genetic algorithms, and grid search to select the parameters such as the number of LSTM units, and the number of neurons, among other parameters.

2. Literature Review
Literature reviews on forecasting models can group various categories according to the types of input parameters (socio-economic factors, environmental factors, customer types, and date), time scales, and resolution (very short-term, less than 1 hour, short-term, 1 hour to 7 days, medium-term, months or seasons, long-term, more than one year). (Kuster, 2017). Statistical models allow forecasting or extrapolating demand in the long-term considering correlations of current consumption with variables that affect demand such as history, climatic variables, economic activities, regions, among others (González Casimiro, 2009). On the other hand, models based on machine learning (ML) and deep learning (DL) have enabled superior forecasts compared to their competitors, on short and very short-term time scales.

DL-based models use artificial neural networks (ANNs). They use nonlinear and stochastic functions to capture the dynamics of the time series from energy consumption and associated exogenous variables (Géron, 2019). They allow information to be extracted from the large amounts of data, historical records available using the computational power provided by graphics processors (Graphics Processors Units). When dealing with time series, models based on recurrent neural networks (RNN) take advantage of previous output information (recurrence) to give a forecast of demand from historical data of power grid usage. The RNNs were improved with the use of Long-Short Term Memory (LTSM) cells to overcome problems caused by null gradients generated during training (Schmidhuber, 1997). The LSTM cell allows you to include exogenous variables and capture important information to extrapolate the data. It contains critical information that has been learned over time, and the network is designed to effectively maintain useful information in it over many time steps. With these new architectures, time series and energy demand forecasting in countries such as Spain and France have been achieved (Cabezón, 2018).

This work explores the effects of hyperparameters such as the number of hidden units and historical window length. The selection and optimization of hyperparameters help to obtain state-of-the-art performance. The number of hidden units is the dimension of the hidden state of the LSTM cell, it represents the model's learning capacity. The historical window refers to the number of input time steps. In our context, the input time steps refer to the number of historical data that the LSTM sees before making a prediction.

3. Data Collection
The dataset considered for this work was provided by the National Electricity Administration (ANDE), it contains the hourly electric demand (in MW, Megawatt) of the national interconnected system of Paraguay (SIN) from the year 2009 to the year 2021.

The meteorological data were obtained from the Global Data Assimilation System (GDAS) which interpolates data from a variety of observing systems and instruments onto a three-dimensional grid. We extracted the air temperature for the most populated region i.e., Asunción. The data from the meteorological stations are recorded every 3 hours, and consequently, a spline interpolation was performed to combine them with the demand data.

There is a strong seasonality in the demand time series, due to the daily activities of the population and a strong increasing trend related to demographic growth and industrialization. Furthermore, average consumption is higher in the warmer months due to the greater use of refrigeration equipment. Likewise, the population's routines are generally reflected in lower consumption on weekends compared to working days, this is reflected in Figure 1.

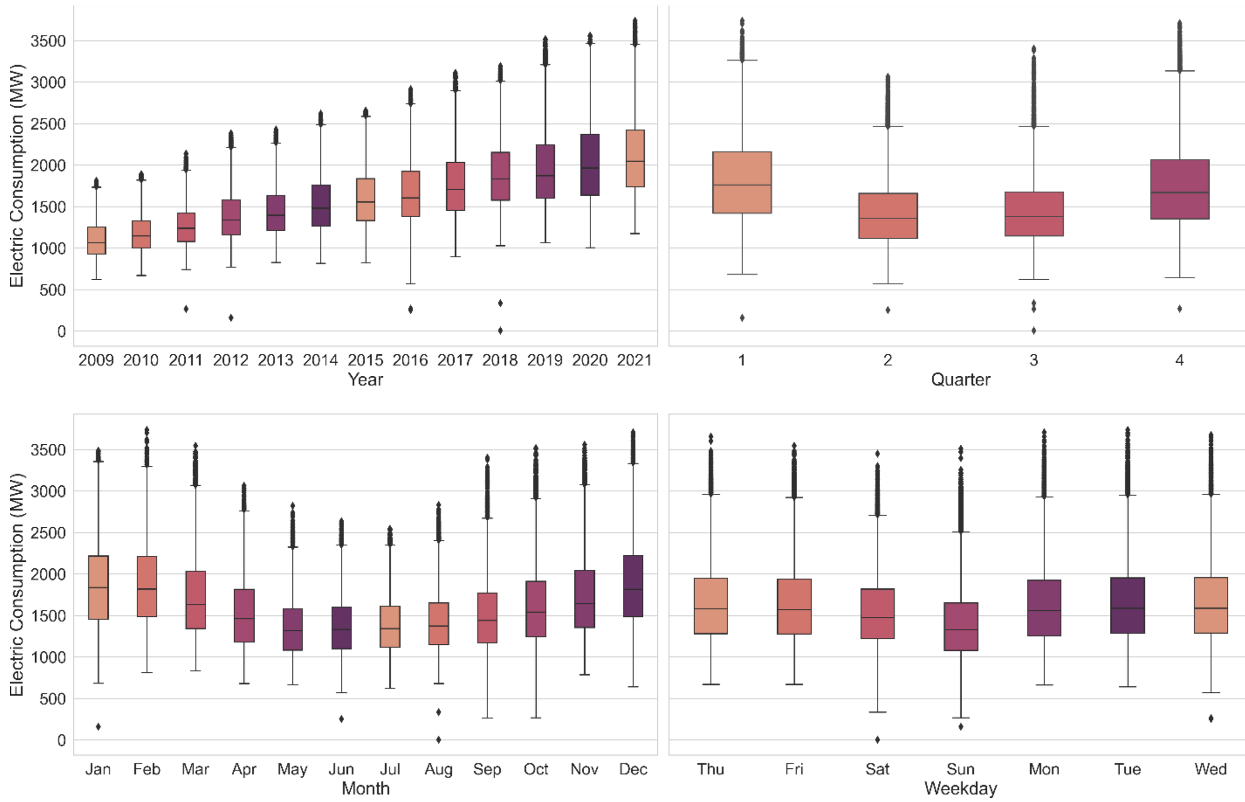

Figure 1. Behavior of electricity consumption over the years, and with respect to quarters, months and days of the week.

A literature review also indicates that it is essential to consider meteorological data that may influence electricity demand. For this reason, the level of influence of temperature in the Asunción area with respect to electricity consumption was measured using Pearson's correlation coefficient r . Thus, as shown in Figure 2, $r = 0.585$, which indicates a direct incidence between this variable and electricity consumption that contrasts with the above mentioned, because this region is the most populated, and also the country is relatively small. (Figure 2)

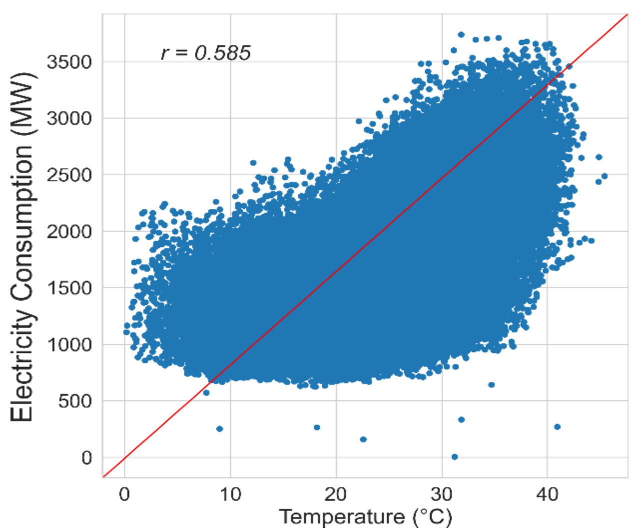

Figure 2. Pearson's correlation coefficient of electric demand vs temperature.

4. Methods

This section presents the model, the input attributes are selected based on the demand data, the calendar variables (day, month, year, among others), and the meteorological data (temperature). The neural network model is then built using recurrent units (LSTM) combined with a fully connected network. These hyperparameters are optimized by considering different approaches such as the Genetic Algorithm, Grid Search, Random Search and Bayesian optimization.

4.1 Prediction Model

The neural network-based predictive model is composed of an input layer with recurrent LSTM cells (Gers, 2000), followed by a single feedforward hidden layer, as shown in Figure 1. The functioning of each LSTM cell is given by Equation 1.

$$\begin{aligned} i_t &= \sigma(U_i \cdot x_t + W_i \cdot h_{t-1} + b_i), \\ f_t &= \sigma(U_f \cdot x_t + W_f \cdot h_{t-1} + b_f), \\ o_t &= \sigma(U_o \cdot x_t + W_o \cdot h_{t-1} + b_o), \\ g_t &= \tanh(U_g \cdot x_t + W_g \cdot h_{t-1} + b_g), \\ c_t &= f_t * c_{t-1} + i_t * g_t, \\ y_t &= h_t = o_t * \tanh(c_t), \end{aligned} \tag{1}$$

where U_i, U_f, U_o and U_g are the weights matrices related to the input vector x_t. W_i, W_f, W_o and W_g are the weights matrices related to the previously hidden state h_{t-1}. Then, b_i, b_f, b_o and b_g are constant, also called bias. The cells employ 3 gates to feed relevant information over time, determined by i_t, f_t and o_t which refer to the input, forget and output gates respectively. In turn, g_t represents the memory cell candidate, which, together with the previous memory cell state c_t, and gates f_t and i_t, determine the new state of the memory cell c_t. Finally, the current output state y_t, or current hidden state h_t is obtained.

The input data are represented as $\boldsymbol{q}$ time series (corresponding to the number of the input features), and the historical window length is given by $\boldsymbol{k}$ i.e., to the number of LSTM cells. Let $\boldsymbol{j}$ be the size of hidden state (h_t), which is feed-forwarded to a fully hidden layer with $\boldsymbol{l} = \boldsymbol{j}$, in order to simplify the model. In turn, the output layer consists of 24 neurons, corresponding to the hourly consumption of a day, from 0 to 23 hrs. Figure 3 shows the LSTM architecture considered in this work. In this work, we focus on the performance optimization of the model and its dependence on the hyper-parameters $\boldsymbol{k}$ and $\boldsymbol{j}$. For the training the loss function used was the mean squared error (MSE) and the ADAM algorithm.

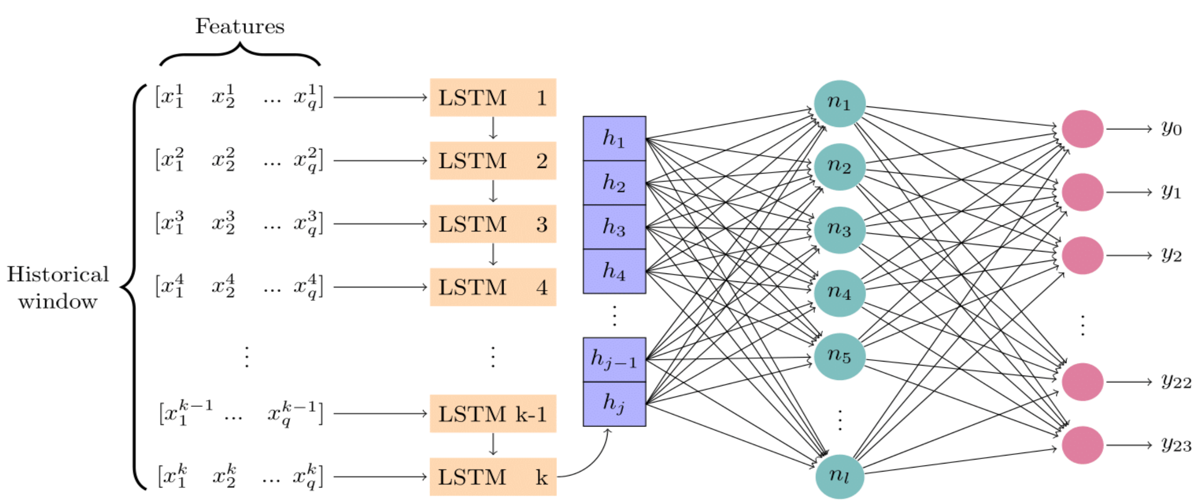

Figure 3. Representation of the architecture of the LSTM cell-based model used for electricity demand forecasting.

The model has several input features, such as the historical electricity demand values, calendar variables (year, month, day of the week), and hour of the day. The hourly temperature of Asunción has also been considered. Days of the week and months are categorized with numerical values. The model access to the previous values demands through the historical windows and the week ago value.

4.2 Optimization Algorithms

The hyperparameters mentioned before need to be optimized to improve the performance of the LSTM model. In this work we apply four different optimization methods which are briefly described below.

1. **Grid search (GS)** is a common technique to find optimum values of hyperparameters by exhaustively searching over all possible parameter combinations. The user defines the search space for each parameter and these choices define a grid where every point represents one combination of the chosen hyperparameters. The method retrains and evaluates the model for each of those combinations and finds the ones with the best metrics. An advantage of this method is that it allows the user to map out the problem space with a sparse grid when needed. Also, each combination can be run independently which allows for parallelization. However, the number of evaluations grows exponentially with the number of hyperparameters considered. This means that GS can be slow to run as it can become computationally expensive as more combinations are required to evaluate.

2. **Random search (RS)** is an alternative to deal to deal with the exhaustive GS method and its high computational cost. The method consists in randomly generating combinations of hyperparameters from the selected search space and find the optimal hyperparameter values for the built model. RS was reported to be more efficient than a GS for hyperparameter optimization (Bergstra J. a., 2012) . A natural limitation of RS is that hyperparameters are chosen randomly in each iteration not considering previous outcomes, which means there is high chance of variance between runs.

3. **Genetic algorithms (GA)** are a set of techniques inspired by the evolution process of natural selection, which serve to optimize a cost a functional cost called fitness. For the initial generation, random individuals are chosen to form a population pool. Individuals with a better performance are favored with higher chances of passing their genes and features on to upcoming generation. This is implemented by using a function to identify the fittest individuals and adding them to the pool in the next generation. Mutations are introduced in each generation by varying genes inherited by an individual at the beginning of the generation. This process is iterated until convergence is attained. The DEAP library is implemented to train the model, since it serves as a framework for rapid prototyping.

4. **Tree of Parzen Estimator (TPE)** is a sequential model-based optimization approach used to search for the optimal hyperparameters. This Bayesian approach focuses on determining the next set of hyperparameters are chosen based on the outcomes of previously chosen sets. This technique (Bergstra J. a., 2011) assigns a probabilistic model to the objective function and updates the model after evaluating until convergence by means of a selection function. While this is implemented in the hyperopt library, the Hyperas library is used to implement this optimization since it easily combines the use of Keras and the capabilities of Hyperopt.

4.3 Performance Evaluation Metrics

We consider the root mean squared errors, mean (max and percentile) absolute percentual errors, and cross-correlation to quantify and characterize the model prediction errors. The root means squared error (RMSE) take the difference between the actual and predicted values in MW. The mean percentage absolute error (MAPE) is an indicator of demand forecast performance that measures the size of the absolute error in percentage terms, which makes it intuitive when analyzing predicted values. The 90th percentile and the maximum value of the absolute percentage error denoted as 90thAPE and Max AEP, respectively, were used to evaluate the dispersion of the absolute error.

5. Results and Discussion

For this work, we used a workstation equipped with an RTX2080 GPU and a Xeon E5-2630V3 processor with 16GB of DDR4 RAM. A Train-Validation-Test split was performed. The train set contains data from 2015 to 2018 and the validation from 2019, representing a ratio between training and validation data of 80/20. With this setup, the typical training time for each model demands between 3 to 5 minutes and less than 1GB of RAM in the GPU.

Table 1 presents the numerical experiment results considering the validation set (2019). Ten executions were done by changing the random seeds for every method; the rows of the table indicate that the GS has the best results, followed by GA. However, it is important to note that the evaluations and run time needed by the GS scheme is more than double that needed by the GA. The GA is based on a stochastic method that simulates the process of evolution where, in each generation, the fittest offspring are considered for the approach of the problem. However, if in a given generation, during the selection process, an individual is considered and it was not altered by any mutation or combination, it will advance to the next generation, with the same fitness value. As a result, the number of evaluations is reduced, since it is only necessary to perform it in the case of new individuals. However, in the case of brute force methods, such as GS, the number of evaluations to be performed depends on the grid defined by the combinations between the values set for each of the elements. This explains the significant difference between the run times and the number of evaluations required for each proposed method. Also, the Genetic Algorithm's performance can vary according to the population's size and generations defined. In this work we have considered two setups: Genetic Algorithm 1 (GA1) comprises a population of 40 individuals and 80 generations, while Genetic Algorithm 2 (GA2) consists of 50 individuals in population, with only 10 generations. When we have fewer evaluations, e.g., GA2 performance is slightly lower than with GA1. Note also that the TPE also has obtained a similarly good result with only a few evaluations. Figures 4 and 5 show the box plots for the metrics specified above, where the distribution of errors from the ten runs can be seen more clearly.

Table 1: Optimization Algorithms Comparison Considering Performance Metrics and the Runtime on the Validation Set i.e., 2019.

Method	Hyperparameters		Evaluation metrics				Evaluations	Run time (hours)
	Historical Windows	Hidden State Size	RMSE (MW)	R2	MAPE %	90thAPE %		
GS	161	164	155.3±5.5	0.85	5.61	11.6	2496	19.75
RS	164	151	158.9±12.4	0.84	5.84	12.1	1000	9,77
TPE	118	139	159.1±6.9	0.84	5,84	12.1	70	2.31
GA1	117	162	158.9±11.1	0.84	5.84	12.1	1112	11.1
GA2	43	136	163.5±10.3	0.83	5.89	12.3	248	1.7

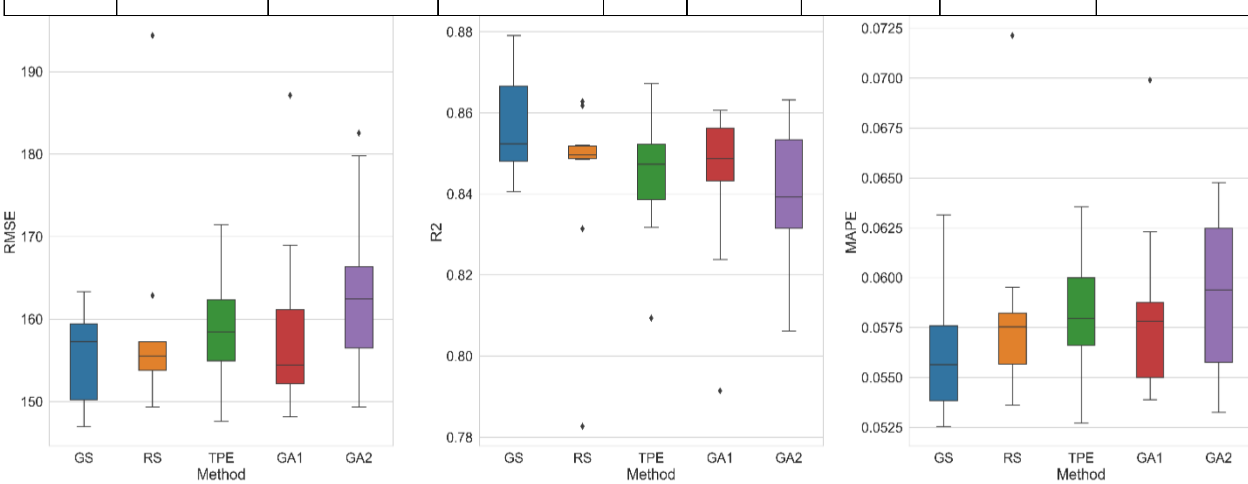

Figure 4: Box Plots Presenting the Performance Comparison considering the RMSE, R2 and MAPE.

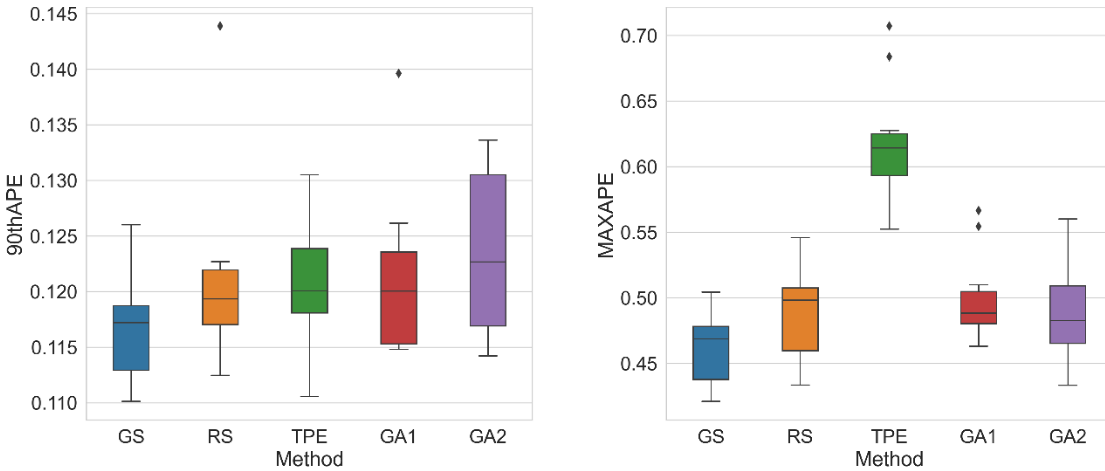

Figure 5: Box Plots Presenting the Performance Comparison considering the 90thAPE and MAXPE

Considering the Covid19 pandemic, we can expect that the demand dynamics changes may affect the models' performance. Table 2 presents the numerical experiment results considering the different training and test sets to analyze the impacts of the pandemics. Ten realizations were made by changing the random seeds. Figures 6 and 7 show that the RMSE and the MAPE are larger when 2020 is included in the training set. Figure 7 also shows that Maximum errors are larger when we try to predict 2020 based on the previous year's information.

Table 2: Best Model Characterization Considering the Test Set.

Cases	Train	Test	Hyperparameters	
			Historical Windows	Hidden State Size
1	2016-2019	2020	161	164
2	2017-2020	2021	161	164
3	2016-2019	2021	161	164

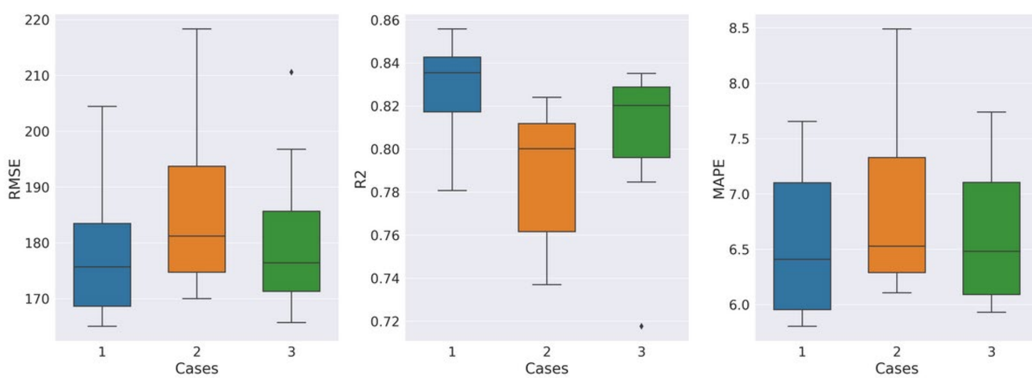

Figure 6: Box Plots Presenting the Performance Comparison considering the RMSE, R2 and MAPE.

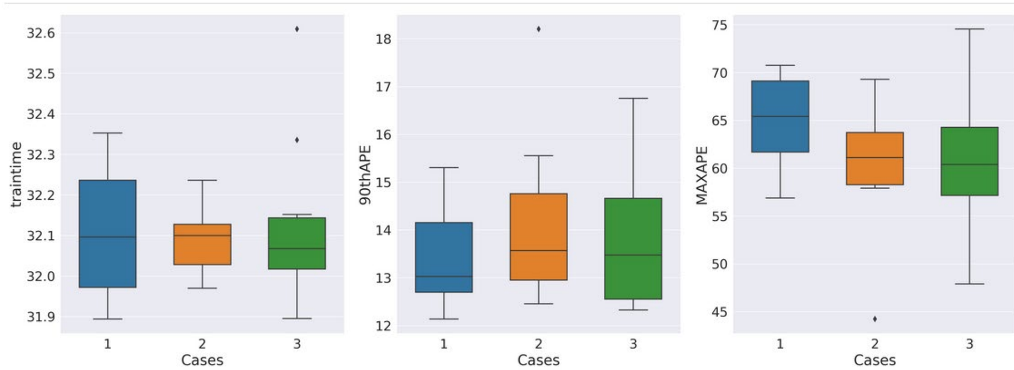

Figure 7: Box Plots Presenting the Performance Comparison considering the training time, 90thAPE and MAXPE

6. Conclusion

In this work, a model based on LSTM neural networks has been developed and characterized to provide a robust 24-hour forecast of the hourly electricity demand for the next day. The influence of exogenous variables with respect to the electricity consumption has been analyzed, resulting that temperature and calendar variables allow an extra contribution to the forecasting problem. The proposed model is constituted for a historical window that allows collecting information from past records. In turn, the dimension of the hidden state and the number of neurons in the hidden layer are determinants for its performance. The choice of these hyperparameters was determined using optimization methods. The methods considered were grid search, random search, genetic algorithm and a Bayesian scheme, called tree of Parzen estimator. Our results indicate that grid search and genetic algorithms can obtain the optimal hyperparameters. However, grid search requires much more evaluations and computational resources to obtain good results. The RMSE and MAPE can vary between 155.3±5.5, and 5.61±0.29 % respectively. The Covid19 pandemic affected the demand dynamics and prediction performance of the model. This model will help power utilities to schedule hourly generation sources for the next day.

7. References

Bergstra, J. a. Algorithms for hyper-parameter optimization. *Advances in neural information processing systems*. (2011).

Bergstra, J. a.. Random search for hyper-parameter optimization. *Journal of machine learning research*. (2012)

Bogado, J., Stalder, D. H., Gómez, S., & Schaerer, C. E. Deep learning-based dengue cases forecasting with synthetic data. *Proceeding Series of the Brazilian Society of Computational and Applied Mathematics. 7*, p. 3. Campinas: SBMAC. (2020).

Bogado, J., Stalder, D. H., Schaerer, C. E., & Gómez-Guerrero, S. ime Series Clustering to Improve Dengue Cases Forecasting with Deep Learning,. *XLVII Latin American Computing Conference (CLEI)* (pp. 1-10). (2021).Ecuador: IEEE. doi:10.1109/CLEI53233.2021.9640130

Bouktif, S. a.. Optimal deep learning lstm model for electric load forecasting using feature selection and genetic algorithm: Comparison with machine learning approaches. *Energies*, 1636. (2018)

Cabezón, M. Predicción demanda eléctrica española. Implementación de redes neuronales recurrentes en Python. (2018).

Cheng, Y. a.. PowerLSTM: power demand forecasting using long short-term memory neural network. In *International Conference on Advanced Data Mining and Applications* (pp. 727-740). Springer. (2017)

Gallardo, J. A., García-Torres, M., Gómez-Vela, F., Morales, F., Divina, F., Becerra-Alonso, D., ... & Román, J. C. M. Forecasting Electricity Consumption Data from Paraguay Using a Machine Learning Approach. In International Workshop on Soft Computing Models in Industrial and Environmental Applications (pp. 685-694). (2021, September). Springer, Cham.

Gardner, F.-M. D.-A. DEAP: Evolutionary Algorithms Made Easy. *Journal of Machine Learning Research*, 2171-2175. (2012).

Géron, A. *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems.* O'Reilly Media. (2019).

Gers, F. A., Schmidhuber, J., & Cummins, F. Learning to forget: Continual prediction with LSTM. *Neural computation*, *12*(10), 2451-2471. (2000).

González Casimiro, M. P. Análisis de series temporales: Modelos ARIMA. *Economía Aplicada III/Ekonomia Aplikatua III, UPV/EHU*. (2009).

Kuster, C. a. Electrical load forecasting models: A critical systematic review. *Sustainable cities and society*, 257-270. (2017).

Morales, F., García-Torres, M., Velázquez, G., Daumas-Ladouce, F., Gardel-Sotomayor, P. E., Gómez-Vela, F., ... & Becerra-Alonso, D. Analysis of Electric Energy Consumption Profiles Using a Machine Learning Approach: A Paraguayan Case Study. Electronics, 11(2), 267. (2022).

Schmidhuber, S. H. Long Short-Term Memory. *Neural Computation*, 1735-1780. (1997).

Seo, C.-h. W. Decomposition and statistical analysis for regional electricity demand forecasting. *Energy*, 313-325. (2012).

Biographies

Félix Morales, received the BS degree in Mechatronics Engineering (2020) from the Engineering School of the National University of Asuncion (FIUNA), San Lorenzo, Paraguay. Currently, he is a part-time researcher at FIUNA in the Mechanics and Energy Laboratory. His research interests include automation systems, data science and artificial intelligence modeling.

Carlos Sauer, received the BS degree in Electronic Engineering (2009) from the Engineering School of the National University of Asuncion (FIUNA), Paraguay, and a Master of Science in Computational and Applied Mathematics from the National Institute of Pure and Applied Mathematics (IMPA), Brazil. Currently, he is a part-time researcher at FIUNA and a member of the consulting team for the Administration Nacional de Electricidal (ANDE), state-owned electric utility in Paraguay. His research interests include mathematical modeling, numerical methods, data analysis and machine learning.

Hans Mersch, is a mechatronics student at the Engineering Faculty of the Asuncion National University (FIUNA). His research interests include time series analysis and deep learning models.

Diego H. Stalder, received the BS degree in Electronic Engineering (2010) from the Engineering Faculty of Asuncion National University (FIUNA), the Master (2013) and the PhD degree (2017) in Applied Computing from the National Institute for Space Research, Brazil. Since 2019, is a full-time researcher at FIUNA, Paraguay. His research interests include time series analysis, deep learning models, biometric signal processing and instrumentation.

Miguel García-Torres, received the BS degree in physics and the PhD degree in computer science from the Universidad de La Laguna, Tenerife, Spain, in 2001 and 2007, respectively. Currently, he is a lecturer in the Escuela Polytechnic Superior of the Universidad Pablo de Olavide, Seville. His research interests include data mining, machine learning, data reduction, bioinformatics, and Astro statistics.