Energy Reports 10 (2023) 3315–3334

Contents lists available at ScienceDirect

Energy Reports

journal homepage: www.elsevier.com/locate/egyr

Research Paper

Comparing Long Short-Term Memory (LSTM) and bidirectional LSTM deep neural networks for power consumption prediction

Davi Guimarães da Silva [a,b,c,*], Anderson Alvarenga de Moura Meneses [a,c]

[a] Federal University of Western Pará – Graduate Program in Society, Nature and Development, R. Vera Paz, s/n, Salé, CEP 68.035-110 Santarém, PA, Brazil
[b] Federal Institute of Education, Science and Technology of Pará, Brazil
[c] Federal University of Western Pará, Institute of Geosciences and Engineering, Laboratory of Computational Intelligence, Brazil

ARTICLE INFO

Keywords:
Electric consumption forecast
Deep learning
Univariate time series
Deep neural networks
Long-Short Term Memory

ABSTRACT

Electric consumption prediction methods are investigated for many reasons, such as decision-making related to energy efficiency as well as for anticipating demand and the dynamics of the energy market. The objective of the present work is to compare two Deep Learning models, namely the Long Short-Term Memory (LSTM) model, and the Bi-directional LSTM (BLSTM) for univariate electric consumption Time Series (TS) short-term forecast model. The Data Sets (DSs) were selected for their different contexts and scales, with the goal of assessing the robustness of the models. Four DSs were used, related to the power consumption of: (a) a household in France; (b) a university building in Santarém, Brazil; (c) the Tétouan city zones, in Morocco; and (d) the aggregated electric demand of Singapore. The metrics RMSE, MAE, MAPE and R^2 were calculated in a TS cross-validation scheme. Friedman's test was applied to normalized RMSE (NRMSE) results, showing that BLSTM outperforms LSTM with statistically significant difference ($p = 0.0455$), corroborating the fact that bidirectional weight updating significantly improves the LSTM performance with respect to different scales of electric power consumption. The present work provides statistical evidence supporting the conclusion that BLSTM outperforms LSTM models according to the tests performed, based on a complete methodology for TS prediction, and also establishes a baseline for future investigation of electric consumption TS prediction.

1. Introduction

Electric load forecasting has great importance in the context of a range of operations. For example, short-term load forecasting is necessary for the stability of power systems as well as for optimal dispatching (Caro et al., 2020). Also, forecast errors imply profit reduction in competitive electricity markets (Bunn, 2000). In the context of energy efficiency (Martinez et al., 2019), electricity consumption forecast is also useful in detecting anomalies, for example, in the behavior of end-users or in faulty appliances (Himeur et al., 2021) as well as for avoiding losses and electric energy waste (Cheng et al., 2022). In this sense, the application of new technologies represents a great opportunity. For example, in 2022 the Federation of Industries of Santa Catarina State (FIESC), from Brazil, reported that the electric energy waste in Brazil is approximately 43 TWh per year (FIESC, 2022), as estimated by the CELESC generation, transmission and distribution company (Centrais Elétricas de Santa Catarina S.A.). Such annual waste is estimated to be equal to consumption by about 20 million Brazilian residences.

Therefore, the development of electric power monitoring systems is extremely important for energy efficiency. In order to cope with energy efficiency challenges related to planning, distribution and consumption, several Artificial Intelligence (AI) techniques have been investigated (Ahmad et al., 2022), and the application of Deep Neural Networks (DNNs; Chollet, 2018) for consumption prediction is particularly interesting. These models have advantages such as capacity of generalization for a massive amount of data (big data) and fast response once a model is trained, with the possibility of application within Internet of Things (IoT) systems (Serpanos and Wolf, 2018), using cloud or edge platforms. For example, Lee et al. (2019) proposed a system with energy consumption prediction based on the Long-Short Term Memory (LSTM) DNN (Hochreiter and Schmidhuber, 1997) using an IoT system with edge computing for collecting real Time Series (TS) data from an office environment. Also, Da Silva et al. (2022) developed an AI prediction module for an IoT energy consumption monitoring framework and, for different TS data sets, comparing Extreme Gradient Boosting (XGBoost)

* Corresponding author at: Federal University of Western Pará – Graduate Program in Society, Nature and Development, R. Vera Paz, s/n, Salé, CEP 68.035-110 Santarém, PA, Brazil.
E-mail address: daviitb@gmail.com (D.G. da Silva).

https://doi.org/10.1016/j.egyr.2023.09.175
Received 30 May 2023; Received in revised form 26 August 2023; Accepted 28 September 2023
Available online 10 October 2023

List of variables

LSTM variables

W_f	Forget gate weight
W_i	Input gate weight
W_c	Cell state weight
W_o	Output gate weight
b_f	Forget gate bias vector
b_i	Input gate bias vector
b_c	Cell state bias vector
b_o	Output gate bias vector
t	Time step
$\widetilde{C}_t$	Candidate for cell state memory
C_t	Cell state (memory)
σ	Sigmoid function
tanh	Hyperbolic tangent function
h_t	Hidden state
x_t	Current input
f_t	Forget gate function
i_t	Input gate function
o_t	Output gate function

BLSTM variables

$LSTM(\cdot)$	Represents all the functions of standard LSTM with the arguments in $(\cdot)$
$g(\cdot)$	Activation function
$\overrightarrow{h}_t$	Hidden state of the forward sequence
$\overleftarrow{h}_t$	Hidden state of the backward sequence
W_{mn}	Weight from m to n
b_n	Bias of a layer n

Sliding window

SW	Number of records in a sliding window
p	Sliding step
n	Number of records in the data set
X_i	ith record of the data set

Performance metrics

$RMSE$	Root Mean Squared Error
$NRMSE$	Normalized Root Mean Squared Error
MAE	Mean Absolute Error
$MAPE$	Mean Absolute Percentage Error
R^2	Coefficient of Determination
n	Total number of points
y_i	Measured value
$\widehat{y}$	Predicted value
$\overline{y}$	Mean value

Friedman test

χ_F^2	Friedman statistics
N	Number of data sets
k	Number of algorithms tested
R_j	Average rank of the models

and Random Forest (RF) algorithms to the LSTM network, which outperformed the previous methods.

The LSTM network is a type of deep Recurrent Neural Network (RNN). RNNs are networks with one or more feedback loops for temporal processing, with two basic uses related to associative memories and input-output mapping networks (Haykin, 2009), with applications to nonlinear prediction and speech processing. In the 1990s, RNNs were trained with Back-Propagation Through Time (BPTT; Rumelhart et al., 1986) or Real Time Recurrent Learning (RTRL; e.g., Robinson and Fallside, 1987; Schmidhuber, 1992), and research on RNNs focused on the relationship between architecture, training and short-term memory structures such as tapped-delay line memory and Gamma memory (see Mozer, 1993; Haykin, 2009). Nevertheless, deeper RNNs are not suitable for training with backpropagation due to errors that rapidly decrease (see *Vanishing Gradients in Recurrent Networks*, in Haykin, 2009; p. 795). Vanishing gradients impair or block the network's learning, since small changes due to distant inputs in time may not influence the learning, therefore making long-term dependencies difficult to learn. This is a condition also known as the *Fundamental Deep Learning Problem* (Schmidhuber, 2015).

Thus, the LSTM network was designed to cope with the vanishing gradient problem. The LSTM overcomes this problem due to its structure, which is the same as the traditional RNN, but with memory blocks replacing summation units in the hidden layer, and in a broad sense, those blocks are recurrently connected subnets (Graves, 2012). LSTM was able to solve problems in which long term memory is necessary, such as context-free languages (Gers and Schmidhuber, 2001) and protein secondary structure prediction (Chen and Chaudhari, 2005).

Schuster and Paliwal (1997) explored the possibility of using past and future information in sequence prediction. The basic idea of Bi-directional RNNs (BRNNs) is assessing context in both directions: whereas one RNN processes the sequence data in one direction, another RNN processes the data in the opposite direction. BRNNs were also applied to secondary protein structure prediction (Baldi et al., 1999). The bi-directional scheme applied to the LSTM networks results in the Bi-directional LSTM (BLSTM) networks. Graves and Schmidhuber (2005) applied BLSTM to the framewise phoneme classification task, outperforming other architectures. Graves et al. (2008) applied BLSTM to unconstrained handwritten recognition.

Successful integration of LSTM to IoT systems for consumption prediction are reported in the literature (e.g., Lee et al., 2019; Da Silva et al., 2021) and other investigations show the ability of LSTM to outperform other models in the electric consumption domain (e.g., Schirmer et al., 2019; Da Silva et al., 2022). Furthermore, the relevance and competitiveness of the LSTM and BLSTM models was demonstrated in several energy consumption prediction scenarios (e.g., Fernández-Martínez and Jaramillo-Morán, 2022; Das et al., 2020). Nevertheless, the growing complexity in DNN architectures and the specificities of real-life contexts and problems in power consumption prediction require investigations of the robustness of the prediction algorithms on a variety of data sets. In this way, investigation is needed in order to provide statistical analysis regarding a comparison between such models over multiple data sets. In the context of the performance of such models, the following research question was formulated: Is there a statistically significant difference between LSTM and BLSTM models over multiple univariate energy consumption data sets?

Studies that compare DNNs for power forecasting do not address the fundamental question, that is, if a certain model may be considered to have better performance regarding several data sets. Besides, the comparison must be assessed statistically, due to the stochastic natures of the observations and of the training of the models. In addition, finding statistically significant differences between models considering several data sets becomes harder since there may be fluctuations in the predictions and metrics that might affect the results. Therefore, one of the main contributions of the present work resides in the testing for statistical evidence in the comparison of models using several data sets.

During the investigation, other factors must also be considered. It is desirable that the time-dependency in the time-series be addressed accordingly. Additionally, for a fair comparison, due to the nature of AI supervised training, it is necessary to select a hold-out data subset for

performance evaluation (i.e., a data subset which was not used during training).

Therefore, in the present work we compared two DNN architectures for univariate energy consumption TS, namely the LSTM (Hochreiter and Schmidhuber, 1997) and BLSTM networks (Graves and Schmidhuber, 2005), addressing the aforementioned issues. In the first step, four data sets were used (household, building, city zones, an island country (Singapore)), that represent different characteristics and scales. Next, the results were evaluated according to a Time Series Cross Validation scheme (TS-CV; Hyndman and Athanasopoulos, 2018), for addressing the time dependency of the predictions. Finally, the evaluation of the networks was performed in a hold-out data set consisting of the last month of data for each data set, using the Friedman test (Friedman, 1937) for comparison of models over multiple data sets (Demšar, 2006; García et al., 2010).

Thus, the main contributions of the present paper are: (i) prediction of multiple electric consumption TS data sets, with different scales and characteristics; (ii) a comparison based on a complete methodology for TS prediction regarding TS-CV, hold-out subsets, and the statistical test between LSTM and BLSTM networks' results; and (iii) a baseline for future univariate electric consumption TS prediction investigation considering other DNN models.

The remainder of the present article is organized as follows. Section 2 presents the related work. In Section 3 the theoretical background is described. The methodology is described in Section 4. The results are presented in Section 5 and discussed in Section 6. Finally, the concluding remarks are presented in Section 7.

2. Related studies

2.1. Systematic review methodology

To perform a systematic review of the literature, a research protocol must first be defined. Therefore, it was decided to use the approach proposed by Kitchenham et al. (2009), which presents a protocol with guidelines that are widely used in research in the area of computing, and which are grouped into three main phases: 1) planning: defines the research question; 2) conduction: search and selection of primary studies, quality assessment, data extraction, summarization and synthesis of results, and also interpretation of results; 3) report: detailing the results obtained.

Therefore, for the scope of this work, the protocol proposed by Kitchenham et al. (2009) was adapted as follows.

1) **Planning:** definition of the research question that was elaborated based on problematization and justification.
2) **Conduct:** *a) search and selection of primary studies:* through digital databases and common search engines used in the study area, namely, IEEEXPLORE, SCIENCEDIRECT, and Google Scholar. Manual searches on the web were also used; *b) quality assessment:* the inclusion and exclusion criteria were considered, in addition to the identification of experimental research, with the presence of variables belonging to the research question; *c) data extraction:* from the analysis of the results of the approaches; d*) summary and synthesis of results:* The experimental research was identified with keywords belonging to the research question and a table was prepared with the synthesis of the selected articles, containing year of publication, models and main contributions; *e) Interpretation of results:* Analysis of the main approaches that perform prediction of STs as well as IoT-based systems.
3) **Report**: results obtained from the systematic review, described in Table 1 and detailed below.

In order to select a publication, it is essential to take into account its relevance, that is, its potential as a source of primary or secondary study. Therefore, for the research questions, the points referring to the analysis

Table 1
Studies of DNNs applied to electric consumption prediction.

Author	Year	Model	Description
Lee et al.	2019	LSTM	The authors detected abnormalities and predicted energy consumption in an office through the integration of edge computing and an LSTM network. Approach: energy consumption prediction based on a LSTM network integrated with edge computing. Data: energy consumption data were collected during six months in an office. Application: two forecast models were implemented (hour and day-ahead) and the RMSE was calculated as 27%.
Kaur et al.	2019	LSTM	A unified LSTM scheme was presented for energy management in smart grids for analyzing and extracting energy patterns related to demand, forecast and peak reduction. Approach: unified scheme based on RNN-LSTM networks for a Smart Grid system. Data: the case study used data from 112 smart homes. Application: used the RMSE and MAPE metrics and the results indicate that RNN-LSTM obtained the lowest RMSE and MAPE errors.
Schirmer et al.	2019	LSTM	Regression methods were evaluated for forecast of residential energy consumption. LSTM outperforms all the algorithms tested. Approach: evaluated the performance of various regression methods for forecasting energy consumption, including LSTM. Data: the authors used the Smart Meters in London data set, which contains data from 5567 households in London. Application: prediction results showed that the LSTM network outperforms all other algorithms, reducing the MAE by up to 26.7%.
Mellouli et al.	2019	LSTM	Four DNN architectures were evaluated for indoor temperature and energy consumption prediction in a cold room. Stacked LSTM is shown to be the most efficient DNN among the models tested. Approach: used LSTM, Convolutional LSTM, Stacked LSTM, and BLSTM to predict inner temperature and energy consumption in a cold room. Data: data set was divided into five use cases for training and testing. Application: results indicated that the Stacked LSTM was the most efficient.
Das et al.	2020	LSTM, GRU and BLSTM	LSTM, BLSTM and GRU were compared for electric load prediction. BLSTM and GRU performed better in longer prediction horizons. Approach: used the LSTM network for Miscellaneous Electric Loads (MEL) and compared it to BLSTM and GRU networks. Data: time series was obtained from an office with capacity for six graduate students in Abu Dhabi, UAE.

(*continued on next page*)

Table 1 (*continued*)

Author	Year	Model	Description
			Application: based on the results of the RMSE and MAE metrics, the authors concluded that the BLSTM network is the most stable model.
Rafi et al.	2021	CNN-LSTM	They carried out a short-term consumption forecast integrating the LSTM and CNN models, considering the advantages of each one. Approach: used a hybrid model (CNN-LSTM) for short-term consumption forecasts and compared it with the LSTM, radial basis functional network (RBFN) and XGboost models. Data: electric load data of Bangladesh electricity system over six years (January 2014 – December 2019). Application: the proposed CNN-LSTM model gives the lowest values of MAE, RMSE and MAPE compared to LSTM, RBFN and XGboost models.
Farsi et al.	2021	CNN-LSTM	They analyzed the performance for short-term load forecasting integrating the LSTM and CNN (CNN-LSTM) models. Approach: used a model called parallel LSTM-CNN Network or PLCNet. Data: sed two data sets, where the first contains the load consumption of the years 2009–2010 from Malaysia (hourly consumption) and the second contains the load consumption of the years 2012–2017 from Germany (daily consumption). Application: the proposed PLCNet model improved the accuracy from 83.17% to 91.18% for the German data and 98.23% accuracy for the Malaysian data with RMSE = 0.031.
Hou et al.	2021	LSTM	LSTM network was applied for residential consumption prediction with load aggregation in a selected household. The proposed method reduces the MAPE in the prediction, being useful for microgrid applications. Approach: applied the LSTM network to an adaptive load aggregation method for residential consumption. Data: used a residential load set from the Smart Grid Smart City (SGSC) project to predict the short-term residential load. Application: results showed that in all cases the proposed method obtains the best MAPE value.
Hadri et al.	2021	LSTM, XGBoost and SARIMA	LSTM, XGBoost, and SARIMA algorithms were evaluated for very-short load forecasting in one data set using three strategies (univariate, multivariate, and multistep). None of the algorithms outperforms the others in all cases. However, LSTM reaches the worst results in all cases. Approach: applied the algorithms XGBoost, LSTM, and SARIMA, in order to empirically evaluate the precision of the prediction. Data: used the Dutch Residential Energy Database (DRED), which contains electric energy consumption data in aggregated and disaggregated levels. Application: the XGBoost was chosen for implantation because it had a better tradeoff in terms of accuracy and computational cost.
Ozer et al.	2021	LSTM	LSTM network was used with transfer learning for load forecasting. The authors applied cross-correlation between time series in order to decide which data set should be used for training a model and transfer learning for prediction of a new data set. Approach: used the LSTM network and Cross Correlation (XCORR) in a transfer learning scheme. Data: used a database and data from a building. Application: RMSE, MAPE, and MAE results showed that the LSTM with transfer learning succeeded in comparison to RF, XGB, and Light Gradient Boosting Machine (LGBM) models.
Da Silva et al.	2022	LSTM, RF and XGBoost	LSTM, RF, and XGBoost were evaluated for consumption prediction in three time series with TS-CV and statistical tests. LSTM achieved the best results in both data sets (respectively with $p = 0.0718$ and $p < 0.0001$). A third data set was used as a test of generalization, and was forecasted with models trained with the previous time series. In this case, LSTM outperformed the other algorithms ($p < 0.0001$). Approach: evaluated the algorithms LSTM, XGBoost, and RF, with TS-CV for consumption forecast. Data: used a data set with consumption data from a residence, another data set with data from a university building, and a third data set with consumption data from Singapore. Application: LSTM network had better performance with a tendency of lower results of RMSE in two data sets (university building and Singapore) and no statistically significant difference in the data set from the residence.
Fernández-Martínez and Jaramillo-Morán	2022	LSTM and GRU	LSTM and GRU networks, with and without Empirical Mode Decomposition (EMD) and Complete Ensemble EMD (CEEMD) preprocessing, were used for day-ahead power consumption univariate and multivariate forecasting in a hospital. LSTM and GRU models achieved similar performance, but the inclusion of EMD and CEEMD consistently improved the results for the multivariate case. Approach: presented a power consumption forecast approach using LSTM and GRU networks. Data: data set contained power consumption and meteorological variables of a medical assistance building. Application: results showed that the best results were obtained by LSTM with

(*continued on next page*)

Table 1 (*continued*)

Author	Year	Model	Description
			preprocessing in the multivariate scenario.
Shin and Woo	2022	LSTM, RF, and XGBoost	LSTM, RF, and XGBoost were evaluated for energy consumption prediction in Korea. Those algorithms were applied to a time series prior and after the COVID-19 pandemic. The best results were achieved by LSTM in the first period and RF in the second period. Approach: compared the algorithms XGBoost, LSTM, and RF electric consumption forecast. Data: used a data set from South Korea that was divided into periods before and after the COVID-19 pandemic (respectively, period 1 and 2). Application: results showed that the LSTM model achieved lower RMSE and MAPE in period 1, whereas RF presented lower RMSE and MAPE in period 2.
Shaqour et al.	2022	DNN, BI-GRU-FCL, GRU-FCL, BLSTM-FCL, and CNN	Prediction of energy consumption aggregation data was performed by DNN, BI-GRU-FCL, GRU-FCL, BLSTM-FCL, and CNN networks. DNN achieved the higher MAPEs for most aggregation levels and BI-GRU-FCL achieved lower RMSEs. Approach: used five different neural networks for prediction. Data: data set of 479 residential dwellings in Osaka, Japan. Application: results showed that DNN reached the higher MAPE for most aggregation levels and Bi-GRU-FCL indicated lower RMSE with 15% faster training and 40% less DNN parameters.
Mubashar et al.	2022	LSTM ARIMA	LSTM, ARIMA, and Exponential Smoothing were compared for short-term load forecasting, using real-world data from twelve households. MAE results show that LSTM outperformed the other two methods. Approach: compared short-term load forecasting performed by LSTM, ARIMA, and Exponential Smoothing. Data: used three months of real-world data from twelve households. Application: results showed that the LSTM presented smaller MAE surpassing the other two methods.

of the title, abstract, keywords, methodology and conclusion will be considered initially, as follows:

1) The first filter aimed to make a preliminary selection of publications, through the application of the search expressions "Energy Consumption Prediction AND LSTM OR Bidirectional LSTM", in three databases, as well as the use of specific filters for each database;
2) In the second filter, the analysis of inclusion and exclusion criteria established in the subitems below was taken into account:
   a) *Inclusion Criteria: (i) publications that primarily or secondarily address studies related to the prediction of electricity consumption; (ii) publications using LSTM and BiLSTM neural networks; (iii) publications that use neural network applications for IoT systems in their approach.*
   b) *Exclusion Criteria: (i) publications whose keywords are absent from the publication and that there is no variation of these keywords (except for plural cases); (ii) publications where the keywords do not appear in the title, abstract, and/or text of the publication; (iii) publications that approach the topic superficially; (iv) publications that are outside the range of the years 2019–2022 for the three databases and for manual search.*
3) As a third filter, we took into account the number of citations in the literature, in addition to reading the abstracts/abstract, methodology and conclusion of each article selected in the previous step.

The first database for carrying out the initial literature survey was IEEEXPLORE (https://ieeexplore.ieee.org), carried out on October 10, 2022, where 271 (two hundred and seventy-one) studies were found. The following additional filters were also applied: deep learning (artificial intelligence), time series, years 2019–2022.

The second database was SCIENCEDIRECT (https://www.sciencedirect.com), searched on October 10, 2022, where 132 (one hundred and thirty-two) papers were found. The following additional filters were also applied: Article type: Research articles, Publication title: Applied Energy, Energy, Neural Networks. Subject areas: Energy. Years: 2019–2022.

The third database searched was Google Scholar (https://scholar.google.com), conducted on October 10, 2022, where 475 (four hundred and seventy-five) papers were found. The following additional filters were also applied: sort by relevance, in any language, any type, including citations, for the years 2019–2022.

In the manual searches on the web, 40 articles were initially selected that were related to the proposed approach and that were not listed in the search engines used, but for which a more in-depth analysis was deemed important. In these searches, some works were also searched that address the use of prediction with IoT systems, as well as prediction with prediction intervals.

Therefore, after the initial survey to select related studies using the search string, 894 articles were found in the three databases and the manual search. Then, the second filter was applied, which is related to reading the summaries/abstract, taking into account inclusion and exclusion criteria. In the inclusion criteria for each database and manual search, the following were selected: (i) IEEXPLORE (13), (ii) SCIENCEDIRECT (6), (iii) Google Scholar (23); (iv) manual search (10). For the exclusion criteria, the following articles were deleted: (i) those with approaches related to forecasting consumption using techniques unrelated to DNNs (198); (ii) articles that predicted consumption with DNNs that did not involve LSTM or BiLSTM (290); (iii) use of LSTM and BiLSTM neural networks in a different context than that of electricity consumption prediction (357).

Finally, the third filter was applied where we sought to select, among the 52 articles obtained in the second filter, the most relevant articles in relation to the number of citations in the literature, in addition to reading the abstracts, methodology and conclusion. Therefore, the data were interpreted by analyzing the internal consistency of the articles, based on the identification of the experiments described, the relationship between the proposed objectives and the results obtained, and the relationship between the methodological elements and the data, resulting in the following selection: (i) IEEXPLORE (5), (ii) SCIENCEDIRECT (4), (iii) Google Scholar (3); (iv) manual search (3). Thus, after applying the third filter, 15 articles were selected from the three databases and manual search, which are detailed in Section 2.2 below.

2.2. Results of the systematic review

Table 1 presents the summary of the main articles related to the present work.

Lee et al. (2019) proposed a system with energy consumption prediction based on a LSTM network integrated with edge computing. For

this purpose, energy consumption data were collected during four months in an office for training, and the analysis was conducted on two months of data. Two forecast models were implemented (hour- and day-ahead). For the hour-ahead forecast, the energy consumption did not present satisfactory precision. Conversely, for the day-ahead forecast, the training errors were reduced as the iterations were increased. Besides a 27% Root Mean Square Error (RMSE), abnormalities could also be detected, and the analysis of electric consumption could be performed.

Kaur et al. (2019) developed a unified scheme based on RNN-LSTM networks for a Smart Grid system, for an integrated approach and data analysis with greater precision. The case study used data from 112 smart homes. The data were pre-processed and decomposed using the High-Order Singular Value Decomposition (HOSVD) for dimensionality reduction. Then, three network models were applied, namely the RNN, the RNN-LSTM and Autoregressive Integrated Moving Average (ARIMA). The authors used the metrics RMSE and the Mean Absolute Percentage Error (MAPE). Results indicate that RNN-LSTM obtained the lowest errors (RMSE = 3.35 and MAPE = 5.21%), The RNN (RMSE = 4.613 and MAPE 17.312%) and ARIMA (RMSE = 4.27 and MAPE 29.18%).

Schirmer et al. (2019) evaluated the performance of regression methods for energy consumption prediction, namely: Linear Regression (LR), Decision Trees (DTs), Neural Networks, RNNs, Gated Recurrent Unit (GRU), and LSTM. For that purpose, the authors used the Smart Meters in London data set, which contains data from 5567 households in London, obtained between November 2011, and February 2014. It was demonstrated that LSTM network overperforms all the other algorithms, reducing MAE in up to 26.7% when compared to LR.

Mellouli et al. (2019) presented an approach which uses four DNN architectures for inner temperature and energy consumption in a cold room: LSTM, *Convolutional* LSTM, *Stacked* LSTM, and BLSTM. The data set was divided into five use cases for training and testing. The results showed that the *Stacked* LSTM was the most efficient for the data used.

Das et al. (2020) described the usage of a LSTM network for Miscellaneous Electric Loads (MEL) and compared it to BLSTM and GRU networks. The time series was obtained from an office with capacity for six graduate students in Abu Dhabi, UAE. The data set ranges from April to November 2017. According to the metrics RMSE and MAE, the three models achieved good results depending on the device and the horizon prediction. The authors concluded that the BLSTM network is the most stable model, for one day- and one week-ahead forecasts.

Rafi et al. (2021) carried out short-term consumption forecasting for the Bangladesh electricity system, integrating the LSTM and CNN models, considering the advantages of each. The motivating issue cited by the authors is that often one model alone does not work well for forecasting electricity consumption. The hybrid model (CNN-LSTM) consists of a CNN module, an LSTM module and a feature-fusion module. The model was validated using the metrics MAE, RMSE, MAPE and R^2. The CNN-LSTM model was compared with LSTM, RBFN and XGBoost approaches. According to the research, the results of the CNN-LSTM model were superior to the models tested in all validation cases.

The LSTM and CNN integration models were used by Farsi et al. (2021) for forecasting consumption data on an hourly basis in Malaysia and a daily one in Germany, which was called the parallel LSTM-CNN Network or PLCNet. For the Malaysian consumption data, they used 2009 load consumption for training and 2010 load consumption for testing. Similarly, in the German dataset, they used 2012–2015 for training and for Test years 2016–2017. According to the authors, the integrated network improved the accuracy from 83.17% to 91.18% for the German data and yielded 98.23% accuracy for the Malaysian data with RMSE = 0.031, reinforcing that the model performs well to be used for short-term consumption forecasting.

Hou et al. (2021) applied the LSTM network to an adaptive load aggregation method for residential consumption. For this purpose, they used a residential load set from the Smart Grid Smart City (SGSC) project to predict the short-term residential load for 50, 100, 150 and 200 randomly selected households during the month of March 2013. The proposed method based on LSTM was compared with the load forecasting method based on SVR, and another one based on BPNN. The results showed that when the total load is predicted directly, the MAPE of LSTM equals only 9.1%, which is lower than traditional methods. When the aggregate load is predicted separately, the MAPE of the LSTM is equal to 8.3%, while the MAPE of the SVR-based method is equal to 11.2%, and the MAPE of the BPNN-based method is equal to 10.2%. In all cases the proposed method obtains the best MAPE value.

Hadri et al. (2021) investigated the algorithms XGBoost, LSTM, and Seasonal Autoregressive Integrated Moving Average (SARIMA), in order to empirically evaluate the precision of the prediction, the execution time, and the computational complexity. For this purpose, the Dutch Residential Energy Database (DRED) was used, which contains electric energy consumption data in aggregated and disaggregated levels, as well as data related to occupancy, indoor temperature, and weather. The authors concluded that none of the algorithms outperforms the others for the three prediction strategies (univariate, multivariate and multistep). For example, XGBoost outperforms the other algorithms in univariate and multistep cases, ARIMA presented the best performance for the multivariate case, while the LSTM model presented the worst results across the three strategies. Thus, XGBoost was chosen for implantation in an IoT platform for real time electricity consumption prediction, because it had a better tradeoff in terms of accuracy and computational cost.

Ozer et al. (2021) used the LSTM network and Cross Correlation (XCORR) in a transfer learning scheme. After data normalization, XCORR was applied between a database and data from a building, to predict which data is more appropriate for training. Then an LSTM model was trained, and transfer learning was performed for another LSTM for the target building data prediction. RMSE, MAPE, and MAE showed that the LSTM with transfer learning succeeded in comparison to RF, XGB, and Light Gradient Boosting Machine (LGBM) models.

Da Silva et al. (2022) evaluated the algorithms LSTM, XGBoost, and RF, with TS-CV in two data sets. The LSTM network had better performance with a tendency of lower results of RMSE in one data set (UCI--household data set; $p = 0.0718$) and showed a statistically significant difference in the other data set (Santarém, Brazil; LABIC-Building data set; $p < 0.0001$). In a third data set (Singapore data set) used for assessing the capacity of generalization, the forecasting was performed with models trained with the previous data sets. In this case, LSTM also achieved the best results ($p < 0.0001$).

Fernández-Martínez and Jaramillo-Morán (2022) presented a power consumption forecast approach for a medical assistance building using LSTM and GRU networks, testing the *Empirical Mode Decomposition* (EMD) as well as the Complete Ensemble EMD (CEEMD). The data set contains power consumption and meteorological variables collected from September 2016 to July 2021. Case studies were conducted for predicting the hourly consumption: a univariate scenario that considers only the whole data set of active energy consumption, and a multivariate scenario with active power, reactive power, temperature and humidity time series. Results showed that LSTM and GRU networks with EMD and CEEMD preprocessing outperformed the models without preprocessing in all cases. The best results were obtained by LSTM with preprocessing in the multivariate scenario (MAPE = 3.51% and RMSE = 55.06).

Shin and Woo (2022) compared the algorithms XGBoost, LSTM, and RF electric consumption forecast in South Korea dividing the data set in two periods: before and after the COVID-19 pandemics (respectively, period 1 and 2). Data of the period ranging from January 1996 and June 2021 were used for analysis, based on the Total Energy Supply (TES), which also contains highly correlated variables for analysis of the predictions. Results showed that the LSTM model achieved lower RMSE and MAPE in period 1, whereas RF presented lower RMSE and MAPE in period 2, while XGB had higher errors in the predictions.

Shaqour et al. (2022) used aggregated power consumption and DNN,

Bi-directional Gated Recurrent Unit with Fully Connected Layers (BI-GRU-FCL), Gated Recurrent Unit with fully connected layers (GRU-FCL), Long Short-Term Memory with Fully Connected Layers (LSMT-FCL), and Convolutional Neural Network (CNN), for prediction. Results show that DNN had higher MAPE for most aggregation levels, and Bi-GRU-FCL indicated lower RMSE with 15% faster training and 40% less DNN parameters.

Mubashar et al. (2022) compared short-term load forecasting performed by LSTM, ARIMA, and Exponential Smoothing using three months of real-world data from twelve households. MAE results show that LSTM outperformed the other two methods.

It is important to highlight that the studies described above did not compare several univariate TS simultaneously for the results obtained by the network models, and therefore it is necessary to use the NRMSE metric due to the different scales existing in each of the data sets as well as to evaluate the robustness of the models through the Friedman and Nemenyi statistical tests.

As mentioned before, the main contribution of the present research is different from previous studies since LSTM and BLSTM models were trained, tested and statistically evaluated considering four energy consumption data sets, with different characteristics, magnitudes and locations, using the same NRMSE metric and the Friedman and Nemenyi statistical tests. The present results may establish a baseline for future tests regarding other network architectures.

3. Theoretical background

3.1. Long-Short Term Memory (LSTM) deep neural networks

The LSTM DNN architecture was introduced by Hochreiter and Schmidhuber (1997), proposing a solution for the vanishing gradient problem, a usual occurrence in RNNs. In a broad sense, LSTM networks have great capacity for retaining past information (long-term memory), whereas keeping the relevance of recent states (short-term memory).

A common LSTM unit is composed of gates (input, forget, and output). The unit is responsible for the network memory, represented by the activation of the weighted sum, and the gates are means of allowing or impeding the flow of information (Hochreiter and Schmidhuber, 1997; Schmidhuber, 2015).

A LSTM is based on the RNN; however, the inner modules have different components. In a LSTM, the cell state is the most important, and aims to transmit information along the network. The information in the cell state is either discarded or modified by the gates. For more details about LSTM architecture see Graves (2012) and Hasan et al. (2019).

Eqs. (1)–(6) are used in the gates, where W_f, W_i, W_c and W_o are weights; b_f, b_i, b_c and b_o are bias vectors; $\widetilde{C}_t$ is the cell memory; and σ is the sigmoid function, according to Hasan et al. (2019).

The forget gate f_t consists of a sigmoid activation function which is applied to the previous hidden state h_{t-1} and the current input x_t for producing a vector (output) where each element is a value between 0 and 1. This layer "decides" which information will be kept or discarded (represented respectively by the values 0 or 1). The output of the forget gate is given by:

$$f_t = \sigma(W_f.[h_{t-1}, x_t] + b_f) \tag{1}$$

The input gate i_t is responsible for which information will be kept in the cell state. The input values are the previous hidden state h_{t-1} and the current input x_t. The input gate uses a sigmoid and the candidate cell state $\widetilde{C}_t$ uses a hyperbolic tangent function (*tanh*) for deciding which information will be used for calculating the cell state C_t. The sigmoid function determines if the current information is important or not, while the *tanh* function returns a value between −1 and +1. Input gate and candidate cell states are respectively given by:

$$i_t = \sigma(W_i.[h_{t-1}, x_t] + b_i) \tag{2}$$

and

$$\widetilde{C}_t = tanh(W_c.[h_{t-1}, x_t] + b_c). \tag{3}$$

In the second part of this step, the resulting values are combined for updating the cell state. The information of the cell state is updated with the multiplication of the current cell state and the output of the forget gate. If f_t is 0, the result is also 0 and the value becomes insignificant. Otherwise, if f_t is 1, it is kept. Afterwards, the cell state is updated according to:

$$C_t = f_t \times \quad C_{t-1} + \mathrm{i}_t \times \widetilde{\mathrm{C}}_t. \tag{4}$$

Finally, the output gate o_t determines the final cell state and the next hidden state h_t. In this gate, the previous hidden state h_{t-1} and current input x_t are the inputs for a sigmoid function and the current cell state C_t is passed by a *tanh* function. Then, the sigmoid output and the *tanh* output are multiplied to determine what information the hidden layer is going to carry, according to:

$$o_t = \sigma(W_o.[h_{t-1}, x_t] + b_o) \tag{5}$$

and

$$h_t = o_t \times tanh(C_t). \tag{6}$$

The structure of gates used by LSTM networks allows the solution of several problems related to sequential models with dependencies, as is the case of electric consumption univariate TS forecasting.

3.2. Bidirectional Long-Short Term Memory (BLSTM) deep neural networks

Schuster and Paliwal (1997) presented the Bi-directional RNNs (BRNNs). Their basic idea is that two independent networks can process the input data in opposing sequences, for data whose start and end are known previously, for example for the phoneme boundary estimation problem (Fukada et al., 1999). One sequence is processed in the usual forward direction (forward state), whereas the sequence is also processed from the end to the beginning, as shown in Fig. 1.

Within a BRNN structure the neurons of a regular RNN are divided in a bidirectional form: one for backward states (negative time direction) and another for forward states (positive time direction). The inputs of the inverse direction are not connected to the results of both states. Thus, using two directions of time, input data from the past and from the future can be used. The BRNN concept was combined with the LSTM structure, and successful applications of the BLSTM were developed (e.g., Graves et al., 2008; Graves and Schmidhuber, 2005, 2008).

According to Sharfuddin et al. (2018), in the implementation of a BLSTM, two LSTM layers are used. One of them is responsible for the past states and the other is responsible for the future states, as shown in Fig. 2.

According to Zhao et al. (2018), the structure of BLSTM can be explained as follows: the network has two hidden layers, the horizontal arrows represent the bidirectional flow on the temporal axis; vertical arrows represent unidirectional flow from the input layer to the hidden layer, and from the hidden layer to the output layer (straight lines). The curved lines are forward and backward unit flows, respectively.

In more detail, Zhao et al. (2018) explains how BLSTM works. Let $S = \{(y^i, x^i)\}_{i=1}^{N}$ represent the set of N samples. For the sample, input x^i has four features: a three-dimensional path and a time clock. The output y^i depends on different tasks. For the hit-miss classification task, y^i has a binary hit-miss value. For the generation task, y^i is the evaluation of the next point x^{i+1}. Furthermore, the author points out that a single BLSTM layer can be concatenated with a direct sequence and an inverse sequence, as demonstrated in the following notation:

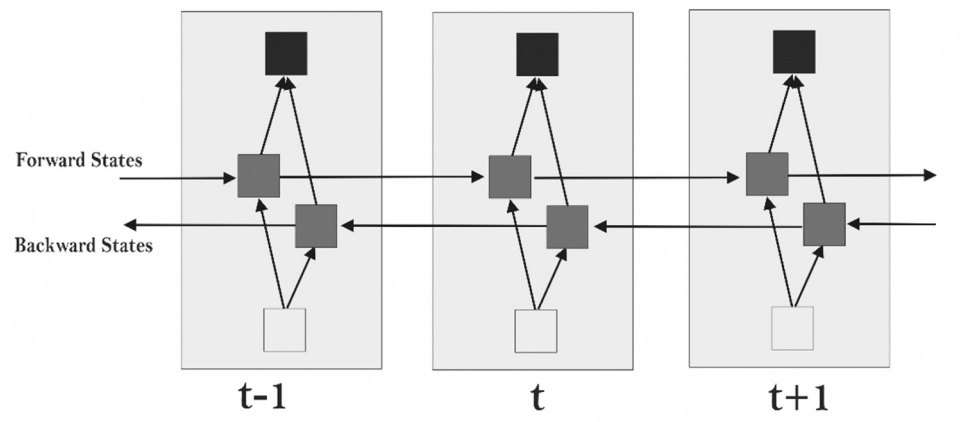


Fig. 1. A BRNN cell structure (Schuster and Paliwal, 1997).

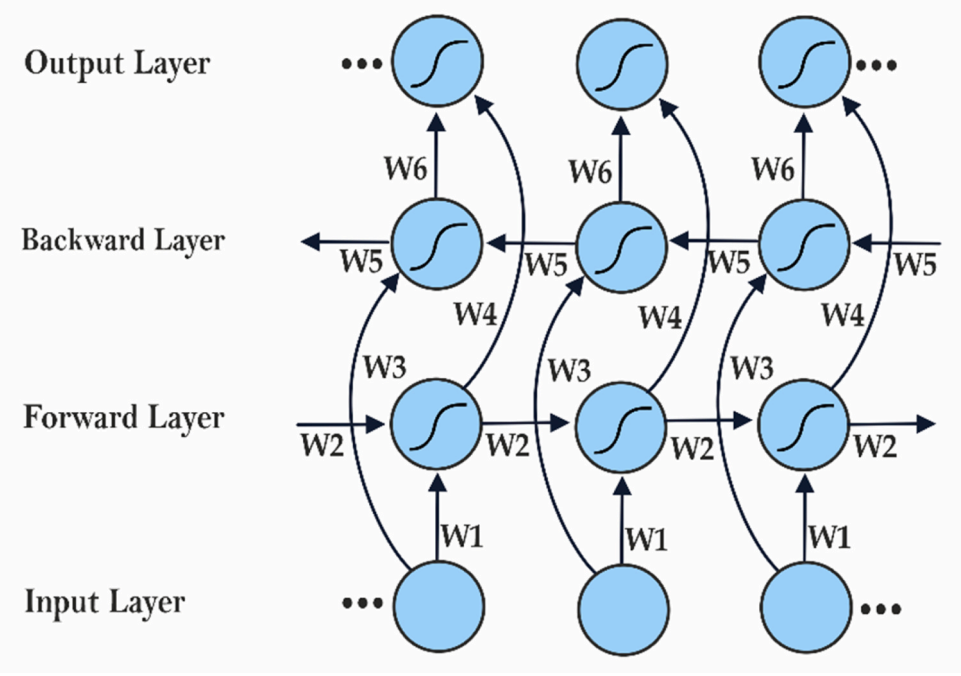


Fig. 2. Example of a BLSTM network model (Sharfuddin et al., 2018).

$$\overrightarrow{h}_t = LSTM(x_t, \overrightarrow{h}_{t-1}) \tag{7}$$

$$\overleftarrow{h}_t = LSTM(x_t, \overleftarrow{h}_{t+1}) \tag{8}$$

$$y_t = g(W_{\overrightarrow{h}y} \overleftarrow{h}_t + W_{\overleftarrow{h}y} \overleftarrow{h}_t + b_y) \tag{9}$$

where, $LSTM(\cdot)$ is used to represent all functions of a standard LSTM, $g(\cdot)$ represents the activation function, W represents the weight and b represents the bias of a given layer.

According to Liang et al. (2020), during the bi-directional stage of the network, the output of the forward LSTM-cell sequence is calculated using inputs in the positive direction while the output of the backward LSTM-cell sequence is calculated using the inputs in the reversed direction. The output of the forward LSTM-cell sequence is calculated using inputs in the positive direction while the output of the backward LSTM-cell sequence is calculated using the inputs in the reversed direction. Later the two outputs are then concatenated and placed in a SoftMax function to normalize their values into a probability distribution, which will produce the final output.

In general, with respect to the difference between the LSTM and BLSTM models, Sharfuddin et al. (2018) stated that while LSTM networks allow the inputs in only one direction, BLSTM allows information flow in both directions, adding a new LSTM layer that inverts the sequence, and the outputs of both layers are combined, for example, with average, sum, multiplication, or concatenation. The possibility of two flow directions enables a better learning process.

4. Methodology

A conceptual diagram of the methodology is shown in Fig. 3. Each DS was preprocessed for extracting the univariate time series (power and time) and downsample for a ten-minute interval. An Exploratory Data Analysis (EDA) was also performed and the TSs graphs, boxplots, and ACF were obtained (Fig. 5). Then, the data were split for obtaining the TS-CV and the holdout subsets. With the TS-CV data, ten LSTM and ten BLSTM models were generated, which in turn were used for prediction of the last month period in each DS and compared to the holdout data. The metrics RMSE, NRMSE, MAE, MAPE, and R^2 were then obtained for

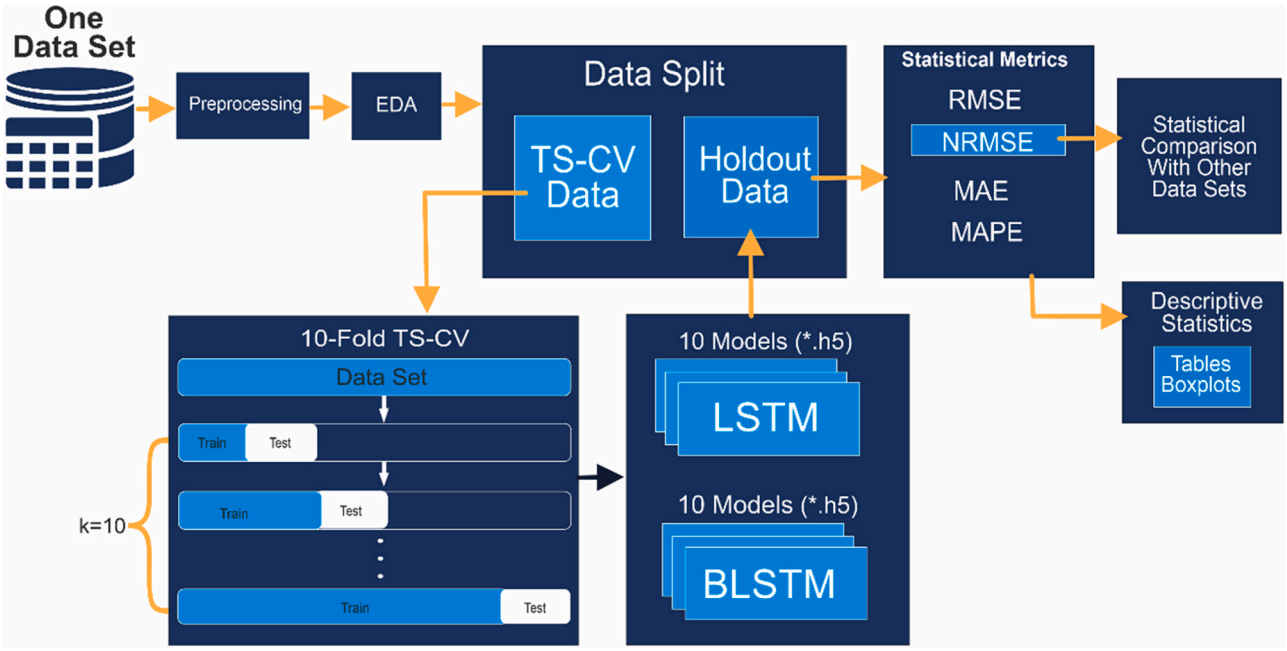


Fig. 3. Conceptual diagram of the methodology.

tables and boxplots. The metric NRMSE was used for comparison among all DSs.

4.1. Case study and data sets

In the present study, experiments were conducted for statistical comparison between LSTM and BLSTM networks with TS-CV. The results were evaluated in four data sets: a) the power consumption of an individual household; b) the power consumption of a university building with classrooms, bathrooms, a library, an auditorium, cafeteria, and administrative departments; c) power consumption in three zones of a city; and d) the power consumption of the island country Singapore, which are described in the next subsections. Table 2 shows a summary of the values for TS-CV and holdout of the data sets.

4.1.1. UCI-household data set

The UCI-Household data set (Hebrail and Berard, 2012) is widely used for electric consumption prediction (e.g., Kim and Cho, 2018; Le et al., 2019; Khan et al., 2021, Gottam et al., 2021). This data set is provided by the *UCI Machine Learning Repository* (Individual household electric power consumption Data Set) and contains data obtained from a household in Sceaux, France. The data were collected from December 2006 to November 2010 (47 months), minute by minute, with a total of 2,075,259 points in kilowatts (kW).

In our investigation, the univariate time series of global active power was used, downsampled to 10-minute intervals (207,526 points). Data from December 2006 to October 2010 (203,801 points) were used for TS-CV and data from November, 2010 (3725 points, 1.79% of the data set) were used as the holdout set.

Table 2
Summary of the TS-CV and holdout subsets for each data set.

Data Sets	Train-Test Set	Holdout Set
UCI-Household	203,801 (98.21%)	3725 (1.79%)
LABIC-Building	29,492 (87.18%)	4338 (12.82%)
Tetouan-Zones	48,096 (91.76%)	4320 (8.24%)
Singapore	20,352 (93.19%)	1488 (6.81%)

4.1.2. LABIC-building data set

The LABIC-Building data set is from a building on the campus of the Federal University of Western Pará (Universidade Federal do Oeste do Pará, UFOPA), city of Santarém, Pará state, Brazil, with a high power demand from air conditioning systems, which is a characteristic of the Amazon region in general (Da Silva et al., 2021, 2022).

The time series contains 256,092 points of aggregated active power in Watts (W). After downsampling the data in 10-minute intervals, the data set had 33,830 points. For the TS-CV there were 29,492 points, ranging from January to July 2019, which corresponds to approximately 87% of the data set. The holdout set was composed of 4338 points from August 2019 (12.82% of the data set).

4.1.3. Tetouan-Zones data set

The Tetouan-Zones data set is related to the power distribution of three zones in Tetouan city, Morocco and was used in several studies of power consumption forecasting (e.g., Salam and El Hibaoui, 2018 ; Singh et al., 2018). The period corresponds to January to December 2017 (12 months), presented in 10-minute intervals with a total of 52, 416 points in kilowatts (kW).

In our research, the period from January to November 2017 (48,096 points) was used for TS-CV, which corresponds to approximately 92% of the data set, and the data points from December 2017 were used as a holdout set (4320 points, 8.24% of the data set).

4.1.4. Singapore data set

The Singapore data set is a time series of electric power demand at a large scale collected from the entire country, which was provided by the website *Energy Market Company Pte Ltd* (EMC; https://www.emcsg.com/marketdata/priceinformation) (EMC, 2010). Its attributes are price type, day, period, price (in $/MW) and demand (MW).

The univariate time series of demand with 30-minute intervals was used. The time period is from January 2010 to March 2011, with 21,840 points. For the TS-CV, 20,352 points were used, ranging from January 2010 to February 2011, which corresponds to approximately 93% of the data set. The holdout set was composed of 1488 points from March 2011 (6.81% of the data set).

4.2. Training and testing

The computational resources were a desktop computer with an Intel® Core™ i5 - 7400 processor with 16 GB RAM, as well as an 11 GB VRAM Nvidia RTX 2080Ti GPU. LSTM and BLSTM networks were implemented in Python using Keras 2.6.0, and Tensorflow 2.6.0 as a backend.

Due to the time dependency of the information, the training data set must be presented in sequence, containing observations in a period prior to the test set. Thus, a TS-CV scheme (Hyndman and Athanasopoulos, 2018; Hewamalage et al., 2023) meets that requirement, and also provides information on the robustness of the models regarding the length of the training set. The *TimeSeriesSplit* function of the scikit-learn was used (see also Da Silva et al., 2022). In the present study, 10 partitions (folds) for all training sets were used ($k = 10$).

In the case of univariate TS, with the TS-CV, it is possible to assess a sensitivity analysis regarding fluctuations in the data sets, through the training of the models on partitioned data with the time-dependency maintained, followed by the observation of the resultant metrics on a hold-out data set, and a statistical test. In other words, TS-CV provides a global performance assessment of the models and verification of their robustness, respecting the time-dependency of the data.

The main parameters of LSTM and BLSTM are described in Table 3.

For each data set in the TS-CV stage, 10 models were obtained for each architecture (LSTM and BLSTM). What differentiates each model is the initialization and the process of adjusting the weights, so it is necessary to perform the statistical tests. The 20 models obtained for each data set were saved in the format ".h5″ for prediction of the hold out subset (1 month data), then performance metrics were generated for comparison. The average Normalized RMSEs (NRMSE) were used for the Friedman test. Repetition and statistical analysis inform about the robustness of the models. With the construction of 10 models, it is possible to evaluate the robustness of the methods, as well as the dispersion of the NRMSE data from the standard deviation value. In addition, there is a need to analyze the robustness and generalization capacity of the network in different time intervals of the same ST. In this way, we can, through statistical analysis, guarantee with a 95% confidence level that the result is statistically significant, and not due to chance.

4.2.1. Sliding window

Dietterich (2002) reviewed the main techniques applied to solve supervised sequential learning problems, and among them the sliding window method stands out. This is because TSs are characterized by being formed by sequential observations, and therefore, it is essential to use a technique that maintains the data collection order while transforming the inputs into a usable format by supervised learning algorithms.

According to Dietterich (2002), the sliding window method converts the sequential supervised learning problem into a classical supervised learning problem. The author points out that this method is the most suitable to be used in TSs data, in view of the possibility of adjusting the model to eventual cycles and trends of the data series.

Table 3
LSTM and BLSTM networks' parameters.

Parameter	Value
MinMaxScaler	feature_range= (0,1)
units	100
epochs	100
batch	32
dropout	0.3
optimizer	rmsprop
Sliding window	90
activation	linear
loss	mse

The implementation of this method requires the definition of three hyperparameters, which are: a) training window size - refers to the number of data points included in a training pass; b) forecast window size - related to the number of data points to be included in the forecast; c) sliding steps - refers to the number of data points skipped from one pass to another.

In terms of practical application, this technique is used by initially defining a training window with SW records and a step p. Then, the model is trained with data from X_1 from X_{sw} advancing a number p of records at each step, continuing until all existing data in the training data set are used. Fig. 4 presents an example of using the sliding window, where the training window has $SW = 90$, step $p = 1$, n represents the number of records in the data set, and forecast window size is equal to 1.

For Dietterich (2002), the use of the sliding window makes the ML algorithms find linear or non-linear combinations between attributes of different times, thus contributing to the improvement of the performance of the algorithms when performing a task.

4.2.2. Additional information

As mentioned before, the data sets have different characteristics and scales (UCI-Household (kW), LABIC-Building (W), Tetouan-Zones (kW) and Singapore (MW)). Thus, to create a Training and Testing dataset with TS-CV, it was necessary to perform the following steps.

1. Initially, data sets were standardized to 10-minute intervals, as some had data collected every 10 s (system default).
2. An exploratory data analysis was also performed.
3. The next step was to implement the LSTM and BLSTM DNNs where the main settings were: MinMaxScaler(0,1), units(100), epochs (100), batchsize(32), dropout(0.3), optimizer(rmsprop), Sliding window(90), activation(linear) and loss(mse). As a general model, the "MinMaxScaler" parameter was used to normalize the consumption values of each data set for the grid input, as they have different scales (W, kW and MW). The RMSprop optimizer was used based on the approach by Reimers and Gurevych (2017). The size of the sliding window with 90 observations was empirically defined for training, testing and validation, being applied to all models, with a view to observing the results obtained by the exploratory data analysis. Other parameters were defined from empirical tests.
4. Then the data were divided into:
   a) TS-CV Data: according to Hyndman and Athanasopoulos (2018) and Hewamalage et al. (2023), TS-CV is an adequate way to evaluate the behavior of a model for TS forecast. Thus, the TimeSeriesSplit function from the Scikit-learn library was used, which performs the partition of a time series into n + 1 parts, where n is the number of partitions informed, seeking to find an optimal number of elements to be used for model training, that is, in each training/test run, the training data is always earlier than the test data, so that the time dependency is respected. The TimeSeriesSplit library was used with 10 partitions (k = 10), as Witten and Frank (2002) from extensive tests performed on different data sets showed that 10 is a value close to the number of partitions in which the best estimates can be obtained. Therefore, for each TimeSeriesSplit fold, the best generated LSTM and BLSTM models were saved in ".h5″ format.
   b) Holdout Data: this is the data that were not trained by the LSTM and BLSTM DNNs, to test the performance of the models (10 models for each DNN) based on performance metrics (Section 4.3). In this case, due to the different scales of each data set, the NRMSE metric was used.
5. After the previous step, the statistical comparison of the NRMSE results between the data sets for the evaluation of networks (LSTM and BLSTM) was carried out in a hold-out data set consisting of the last month of data for each data set, using the Friedman test (Friedman, 1937) for comparison of models over multiple data sets (Demšar, 2006; García et al., 2010).

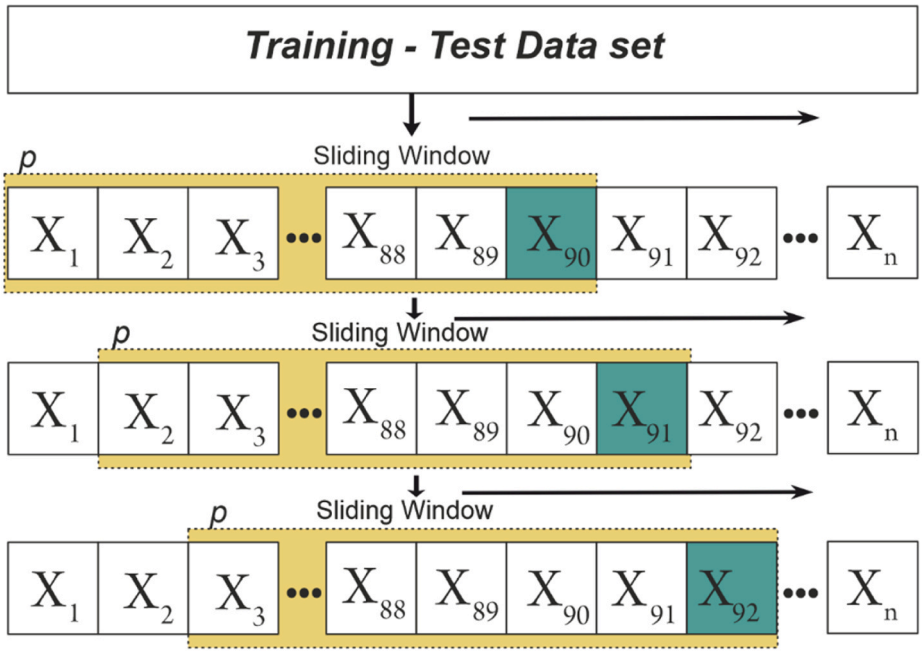


Fig. 4. Training-test example with sliding window.

6. Finally, the results of all metrics for each data set and network (LSTM and BLSTM) are presented in tables, as well as in boxplots that enable visualization of the data distribution, in this case based on the NRMSE metric.

Regarding the long time period needed for training the LSTM and BLSTM with TS-CV (Section 5.4), it is important to highlight that after deciding on the best architecture, the trained model can be saved in ". h5″ format for predictions at any time, requiring much less computational effort than the training. In addition, re-training of the network with additional data can be carried out while the previously saved model is loaded and deployed as a software module (see Da Silva et al., 2022).

4.3. Performance metrics

Five metrics were used for evaluating the models' performance, namely, RMSE, NRMSE, MAE, MAPE, and the coefficient of determination R^2. The NRMSE is the metric used for comparison of data sets with different scales with Friedman's test, since it uses normalization. Such metrics are described by:

$$RMSE = \sqrt{\left[\frac{1}{n}\sum_{i=1}^{n}(|y_i - \widehat{y}_i|)^2\right]} \tag{10}$$

$$NRMSE = \frac{RMSE}{\overline{y}} \tag{11}$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \widehat{y}_i| \tag{12}$$

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - \widehat{y}_i}{y_i}\right| * 100 \tag{13}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \widehat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \overline{y_i})^2} \tag{14}$$

where y_i, $\widehat{y}_i$, and n are respectively the measured value, the predicted value and the total number of points; $\overline{y}$ is the mean value of the time series. Hewamalage et al. (2023) provide more details on performance metrics and other topics in forecast evaluation.

4.4. Statistical analysis

The Friedman test was applied (Friedman, 1937; García et al., 2010) for comparing models obtained in the TS-CV scheme. With the Friedman test, it is possible to determine if there is statistical significance difference considering multiple data sets. The number of 10 NRMSE values for each model in each data set, as suggested by Witten and Frank (2002), makes use of a non-parametric test such as Friedman's appropriate, which is based on the ranks of the NRMSE results.

The first step of the Friedman test is converting the original results to ranks. Such ranks vary from 1 to k, where k is the number of models tested. Thus, the k models are classified according to each data set N separately. For tied scores, an average rank is attributed (see also Demšar, 2006).

The Friedman statistic χ_F^2 is then calculated according to

$$\chi_F^2 = \frac{12N}{K(K+1)}\left[\sum_j R_j^2 - \frac{k(k+1)^2}{4}\right] \tag{15}$$

with k-1 degrees of freedom, and R_j is the average rank of the models (García et al., 2010).

The null hypothesis of the test is that there is no statistically significant difference between models. A p-value p is calculated and the threshold 0.05 was adopted. If $p > 0.05$, the null hypothesis is rejected and at least one of them differs from the others. In our case the Nemenyi test was also applied (Nemenyi, 1963; Barrow et al., 2013). The statistical tests were performed in R (https://www.r-project.org/), with the functions "friedman.test" and frdAllPairsNemenyiTest.

5. Results

5.1. Exploratory data analysis

The time series, quarterly box plots, and ACFs are shown in Fig. 5. The UCI-Household, LABIC-Building, and Singapore data sets were

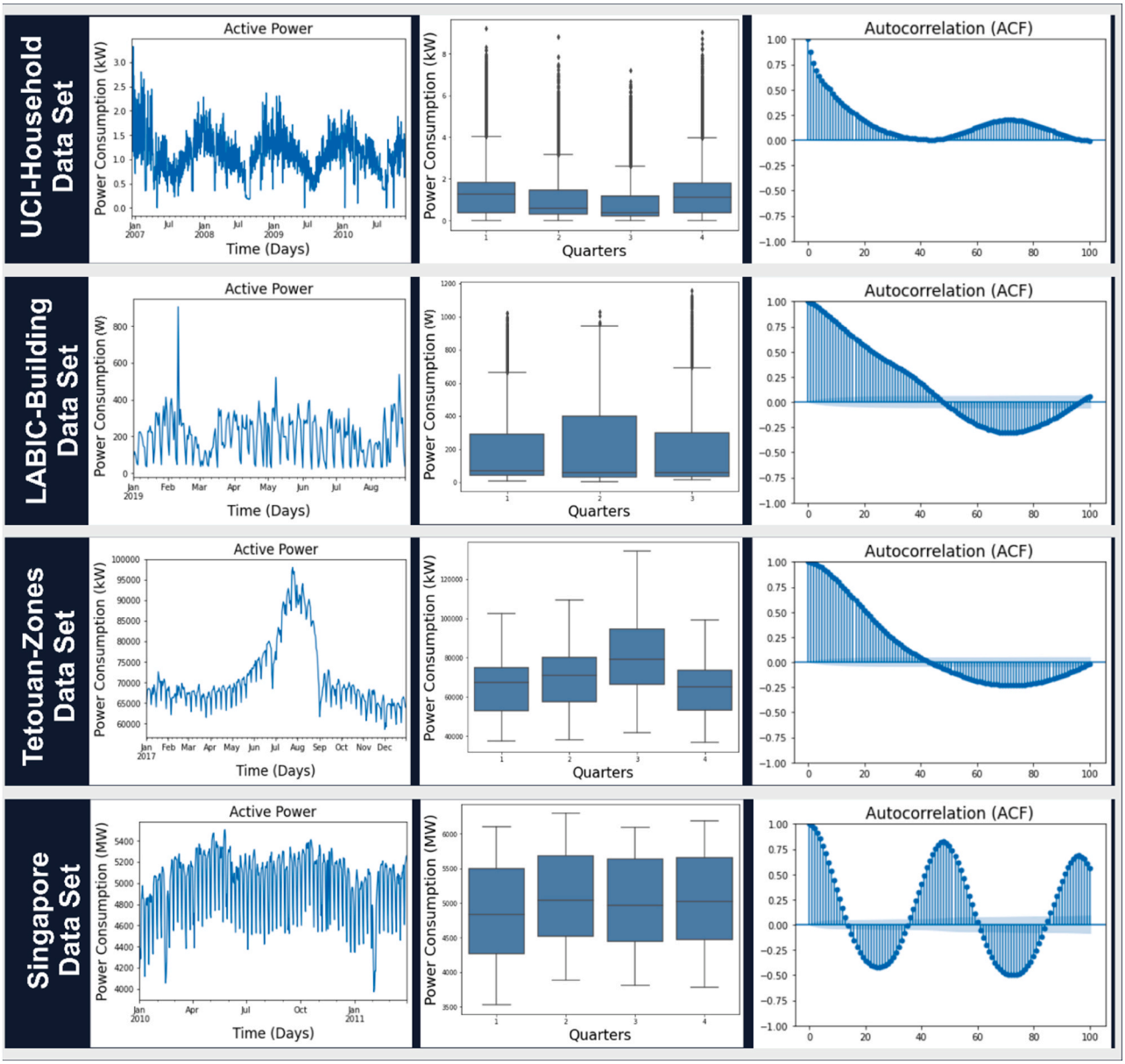


Fig. 5. Power consumption time series; Quarterly boxplots; and Auto-Correlation Functions (ACFs) respectively for the four data sets used in the present work.

analyzed previously by Da Silva et al. (2022), while the Tetouan-Zones data set was included in the present work. The data sets represent different patterns of consumption, contexts, lengths, and scales, chosen to test the robustness of the algorithms.

As examples of the data sets' characteristics, in the UCI-Household data set, in the second and third quarters the power consumption is lower than the first and fourth quarters due to the electric load related to high demand in winter. In the Labic-Building, the second quarter (March to May) represents a period with higher consumption by air conditioning systems.

Another noticeable characteristic of the data sets, in the plotted ACF, a small blue region close to the horizontal axis represents the values with no statistical significance for ACF. In other words, until lag 100, almost all lags show statistically significant auto-correlations.

5.2. Computational results

In the next subsections, the results are presented for the LSTM and BLSTM networks for the UCI-Household, LABIC-Building, Tetouan-Zones, and Singapore data sets, along with the holdout data sets. The results of the statistical tests also will be described, as well as the TS-CV execution times.

5.2.1. LSTM and BLSTM results for holdout sets

5.2.1.1. UCI-household data set. Tables 4 and 5 show the metrics RMSE, NRMSE, MAE, MAPE, and R^2 for 10 models each (obtained by TS-CV), respectively for LSTM and BLSTM models applied to the UCI-Household holdout subset. Fig. 6 shows the NRMSE boxplots for both models. Fig. 7 shows a comparison between typical predictions and the real values of the holdout data set.

5.2.1.2. LABIC-building data set. Tables 6 and 7 show the metrics RMSE, NRMSE, MAE, MAPE, and R^2 for 10 models each (obtained by TS-CV), respectively for LSTM and BLSTM models applied to the LABIC-Building holdout subset. Fig. 8 shows the NRMSE boxplots for both models. Fig. 9 shows a comparison between typical predictions and the real values of the holdout data set.

Table 4
LSTM models' results for the UCI-Household holdout subset.

LSTM Model	Metrics - LSTM models - UCI-Household holdout				
	RMSE	NRMSE	MAE	MAPE (%)	R^2
1	0.484	0.065	0.272	27.3	74.4
2	0.481	0.064	0.266	25.6	74.7
3	0.477	0.064	0.275	28.9	75.1
4	0.594	0.079	0.394	51.2	61.4
5	0.489	0.065	0.270	25.4	73.9
6	0.620	0.083	0.414	50.9	58.0
7	0.548	0.073	0.349	41.9	67.2
8	0.476	0.064	0.270	27.1	75.2
9	0.480	0.064	0.268	27.4	74.8
10	0.605	0.081	0.407	50.6	59.9
Average	0.525	0.070	0.318	35.6	69.5
Median	0.486	0.065	0.273	28.1	74.1
St-Dev	0.060	0.008	0.065	11.6	7.1
Min	0.476	0.064	0.266	25.4	58.0
Max	0.620	0.083	0.414	51.2	75.2

Table 5
BLSTM models' results for the UCI-Household holdout subset.

BLSTM Model	Metrics - BLSTM models - UCI-Household holdout				
	RMSE	NRMSE	MAE	MAPE (%)	R^2
1	0.497	0.066	0.291	30.5	72.9
2	0.510	0.068	0.286	27.3	71.5
3	0.506	0.068	0.290	27.6	72.0
4	0.506	0.068	0.290	27.6	72.0
5	0.499	0.067	0.307	36.2	72.7
6	0.488	0.065	0.286	31.2	74.0
7	0.491	0.066	0.266	24.0	73.6
8	0.499	0.067	0.287	24.1	72.8
9	0.485	0.065	0.265	23.6	74.3
10	0.522	0.070	0.391	59.7	70.2
Average	0.500	0.067	0.296	31.2	72.6
Median	0.499	0.067	0.288	27.6	72.8
St-Dev	0.011	0.001	0.036	10.7	1.2
Min	0.485	0.065	0.265	23.6	70.2
Max	0.522	0.070	0.391	59.7	74.3

5.2.1.3. Tetouan-Zones data set. Tables 8 and 9 show the metrics RMSE, NRMSE, MAE, MAPE, and R^2 for 10 models each (obtained by TS-CV), respectively for LSTM and BLSTM models applied to the Tetouan-Zones holdout subset. Fig. 10 shows the NRMSE boxplots for both models. Fig. 11 shows a comparison between typical predictions and the real values of the holdout data set.

5.2.1.4. Singapore data set. Tables 10 and 11 show the metrics RMSE, NRMSE, MAE, MAPE, and R^2 for 10 models each (obtained by TS-CV), respectively for LSTM and BLSTM models applied to the Singapore holdout subset. Fig. 12 shows the NRMSE boxplots for both models. Fig. 13 shows a comparison between typical predictions and the real values of the holdout data set.

5.2.2. Summary of LSTM and BLSTM results

Table 12 shows the consolidated results (average NRMSE and R^2 *score*) for LSTM and BLSTM networks, applied to the data sets UCI-Household, LABIC-Building, Tetouan-Zones, and Singapore. BLSTM models obtained lower NRMSE averages in all data sets. The statistical testing conducted on the results will be described in the next subsection.

5.3. Statistical tests

In order to verify if there was statistical significance of the results, the Friedman test was applied considering the average NRMSE obtained by LSTM and BLSTM for all four data sets (Table 12).

According to Friedman test, there is a statistically significance difference between LSTM and BLSTM ($p = 0.0455$), and the Nemenyi test also indicated a statistically significant difference ($p = 0.046$). According to these results, BLSTM outperformed LSTM for the time series prediction of electric power consumption.

5.4. TS-CV execution times

The TS-CV execution times (with $k = 10$) for obtaining the trained models are shown in Table 13. The execution times may be compared with the values in Fig. 14, and using the hyperparameters shown in Table 2, BLSTM's TS-CV took more time than LSTM's.

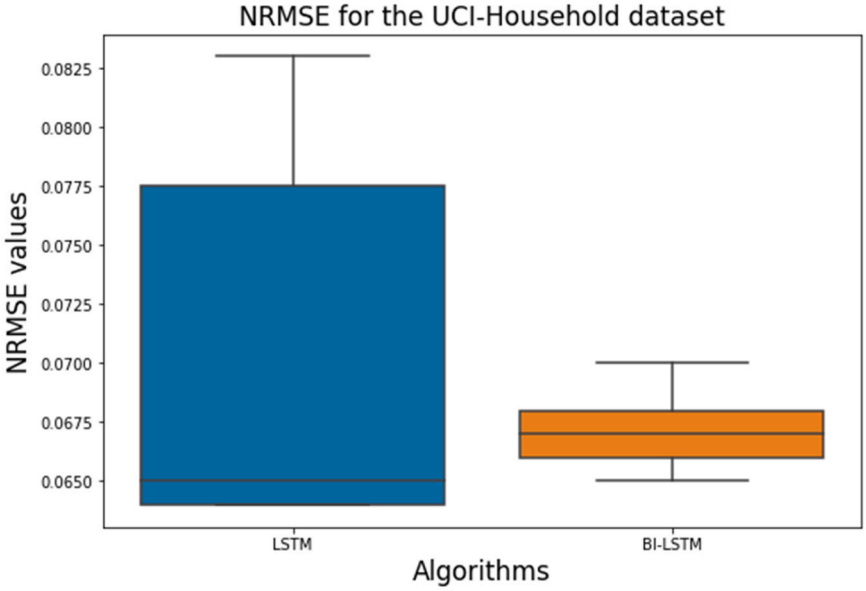


Fig. 6. NRMSE boxplots for LSTM and BLSTM models applied to the UCI-Household holdout subset.

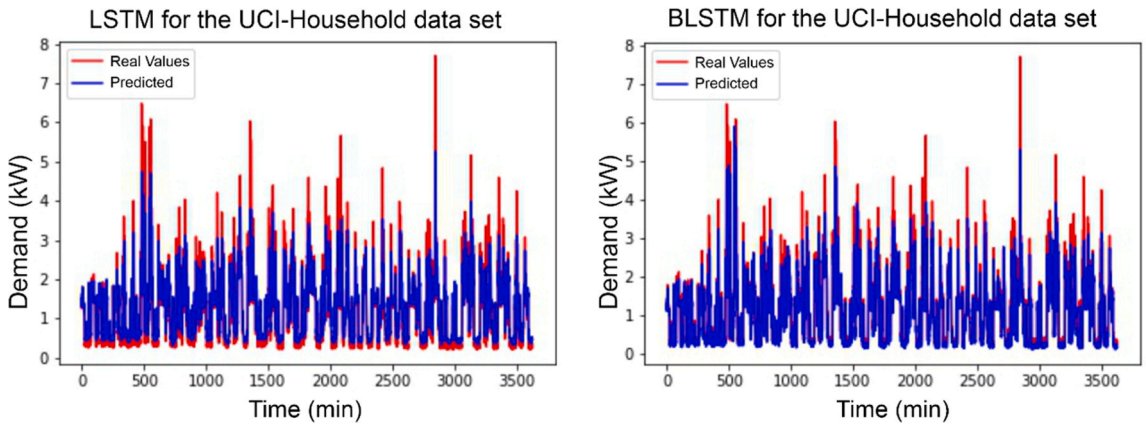


Fig. 7. Typical LSTM and BLSTM predictions for the holdout set (UCI-Household data set).

Table 6
LSTM models' results for the LABIC-Building holdout subset.

LSTM Model	Metrics - LSTM models - LABIC-Building holdout				
	RMSE	NRMSE	MAE	MAPE (%)	R^2
1	35.883	0.032	18.079	12.8	97.8
2	36.193	0.032	22.095	24.4	97.8
3	37.972	0.033	18.648	11.3	97.6
4	35.352	0.031	17.933	14.3	97.9
5	35.065	0.031	17.912	13.1	97.9
6	37.042	0.033	22.176	22.1	97.7
7	35.660	0.031	18.113	13.6	97.9
8	36.060	0.032	19.389	15.9	97.8
9	36.158	0.032	20.287	17.3	97.8
10	34.809	0.031	17.981	13.5	98.0
Average	36.020	0.032	19.261	15.8	97.8
Median	35.971	0.032	18.380	13.9	97.8
St-Dev	0.936	0.001	1.697	4.3	0.1
Min	34.809	0.031	17.912	11.3	97.6
Max	37.972	0.033	22.176	24.4	98.0

Table 7
BLSTM models' results for the LABIC-Building holdout subset.

BLSTM Model	Metrics - BLSTM models - LABIC-Building holdout				
	RMSE	NRMSE	MAE	MAPE (%)	R^2
1	36.587	0.032	22.003	22.6	97.8
2	39.469	0.035	27.645	36.5	97.4
3	36.895	0.032	18.588	12.4	97.7
4	37.574	0.033	25.741	32.1	97.6
5	34.093	0.030	17.225	11.9	98.0
6	34.116	0.030	19.217	18.5	98.0
7	33.521	0.030	16.427	11.0	98.1
8	34.342	0.030	17.134	11.2	98.0
9	34.413	0.030	20.092	21.1	98.0
10	35.989	0.032	21.705	22.0	97.8
Average	35.700	0.031	20.578	19.9	97.9
Median	35.201	0.031	19.654	19.8	97.9
St-Dev	1.925	0.002	3.746	8.9	0.2
Min	33.521	0.030	16.427	11.0	97.4
Max	39.469	0.035	27.645	36.5	98.1

6. Discussion

In the present work the LSTM and BLSTM models generated by TS-CV were evaluated in four electric energy consumption holdout data sets. Each data set represents different scales and consumption characteristics (household, building, city zones, and country). Also, a statistical comparison of the results was performed with the Friedman and Nemenyi tests, indicating that BLSTM statistically outperformed LSTM.

One of the possible reasons for the BLSTM had lower RMSE values in relation to the LSTM is related to the information flow in both directions, enabling better learning process by the neural network, because according to Jurafsky and Martin (2000), the goal of a BLSTM is to gain more knowledge regarding a given context by capturing it from more than one perspective, and then concatenating both outputs into a single contextual representation.

This trend corroborates the results obtained by the approach of Graves and Schmidhuber (2005), as they concluded that bidirectional networks outperformed unidirectional ones. In addition, Siami-Namini et al. (2019), who compared LSTM and BLSTM performance for TS prediction, showed that BLSTM obtained lower RMSE values.

Another comparison is reported by Fang and Yuan (2019), who evaluated DNN models for TS forecasting using LSTM and BLSTM. In the same way, their results pointed to better BLSTM performance regarding the metrics MAE, MAPE, and RMSE. Rhif et al. (2020) also compared LSTM and BLSTM for TS prediction and reported that for the metrics RMSE and R^2, BLSTM was the best DNN in their tests. Das et al. (2020) also compared LSTM, BLSTM and GRU for electric load prediction and reported that BLSTM and GRU performed better for longer prediction horizons.

For execution times, the greater amount of time necessary for training BLSTM (approximately 2.0, 1.6, 1.7, and 1.7 times higher than for LSTM, respectively for each data set) compensates by having higher quality prediction results which were statistically confirmed.

It is important to notice that although other studies present comparisons between algorithms, to the best of our knowledge comparisons of performance considering several data sets with statistical assessment are scarce. It is well known that an algorithm may have better performance in one data set, but poor performance in others, so that a claim of better performance for a specific algorithm must be based on sound methodology with adequate statistical tests, which was the general objective of the present study and was demonstrated for the tested data.

Therefore, the results of the present study fulfill the proposed objectives: (i) four electric consumption TS data sets (with different scales and characteristics) were predicted for comparison between LSTM and BLSTM with statistical evidence supporting that BLSTM outperforms LSTM models; (ii) the comparisons were based on a complete methodology for TS prediction regarding TS-CV, as well as hold-out subsets; and (iii) a baseline for future investigation of univariate electric consumption TS prediction was established.

With respect to future studies, other models such as the Reservoir Computing model (Bianchi et al., 2020; Moon et al., 2019), the CNN-LSTM model (Farsi et al., 2021; Shao et al., 2020), and the CNN-BLSTM model (Bohara et al., 2022; Jogunola et al., 2021) could be compared for the data sets used in this study; in addition, new electricity

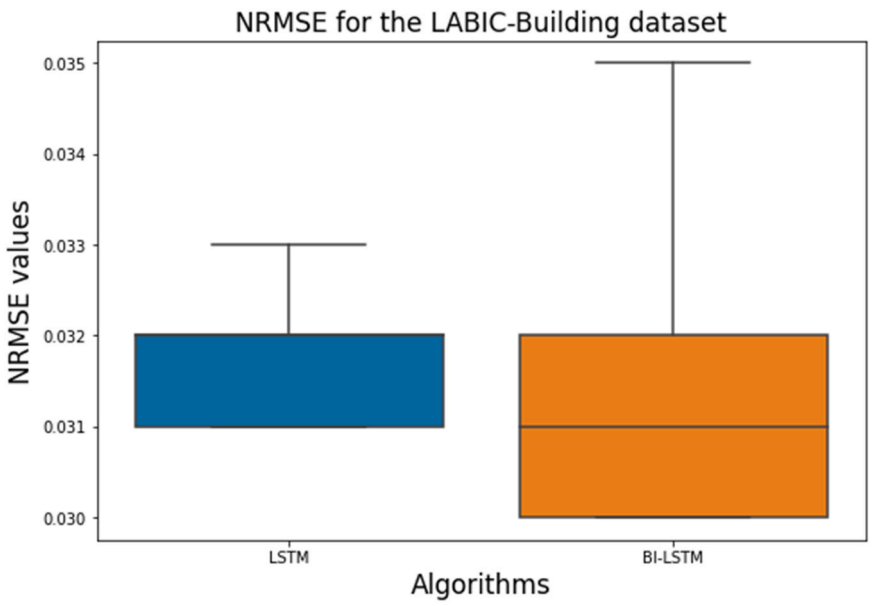


Fig. 8. NRMSE boxplots for LSTM and BLSTM models applied to the LABIC-Building holdout subset.

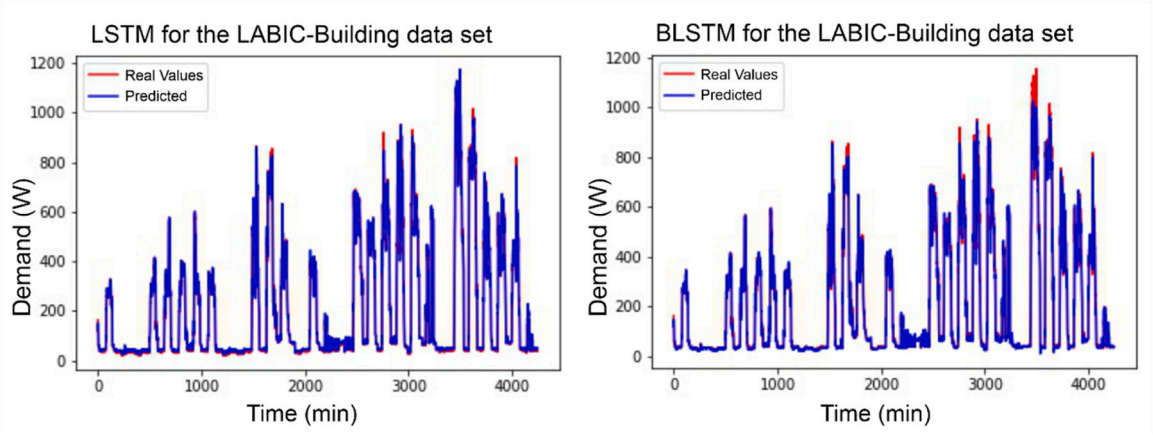


Fig. 9. Typical LSTM and BLSTM predictions for the holdout set (LABIC-Building data set).

Table 8
LSTM models' results for the Tetouan-Zones holdout subset.

LSTM Model	Metrics - LSTM models - Tetouan-Zones holdout				
	RMSE	NRMSE	MAE	MAPE (%)	R^2
1	880.601	0.016	707.742	1.2	99.6
2	806.638	0.015	591.798	0.9	99.7
3	747.032	0.014	569.353	1.0	99.7
4	1264.597	0.023	1013.582	1.5	99.2
5	832.836	0.015	623.614	0.9	99.7
6	785.838	0.014	612.084	1.0	99.7
7	952.206	0.017	807.024	1.4	99.6
8	833.170	0.015	648.251	1.0	99.7
9	810.916	0.015	637.257	1.1	99.7
10	654.841	0.012	472.449	0.8	99.8
Average	856.867	0.016	668.315	1.1	99.6
Median	821.876	0.015	630.436	1.0	99.7
St-Dev	163.161	0.003	149.273	0.2	0.2
Min	654.841	0.012	472.449	0.8	99.2
Max	1264.597	0.023	1013.582	1.5	99.8

Table 9
BLSTM models' results for the Tetouan-Zones holdout subset.

BLSTM Model	Metrics - BLSTM models - Tetouan-Zones holdout				
	RMSE	NRMSE	MAE	MAPE (%)	R^2
1	618.613	0.011	457.710	0.8	99.8
2	576.300	0.010	429.827	0.7	99.8
3	834.071	0.015	641.204	1.2	99.7
4	850.158	0.015	653.708	1.0	99.6
5	730.202	0.013	579.820	1.0	99.7
6	648.534	0.012	514.702	0.8	99.8
7	573.357	0.010	407.655	0.6	99.8
8	523.353	0.009	386.385	0.6	99.9
9	581.298	0.011	432.836	0.7	99.8
10	526.451	0.010	380.853	0.6	99.9
Average	646.234	0.012	488.470	0.8	99.8
Median	599.956	0.011	445.273	0.7	99.8
St-Dev	119.514	0.002	103.106	0.2	0.1
Min	523.353	0.009	380.853	0.6	99.6
Max	850.158	0.015	653.708	1.2	99.9

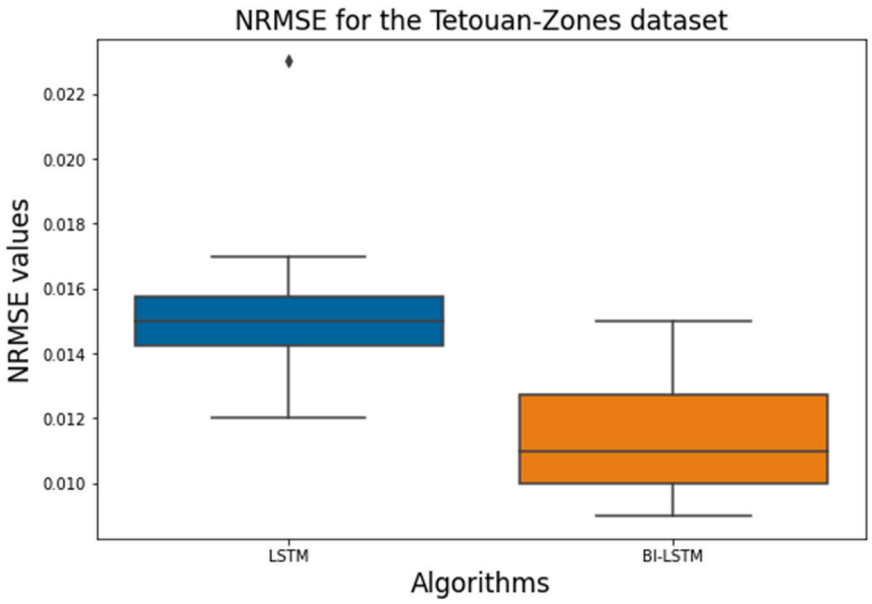


Fig. 10. NRMSE boxplots for LSTM and BLSTM models applied to the Tetouan-Zones holdout subset.

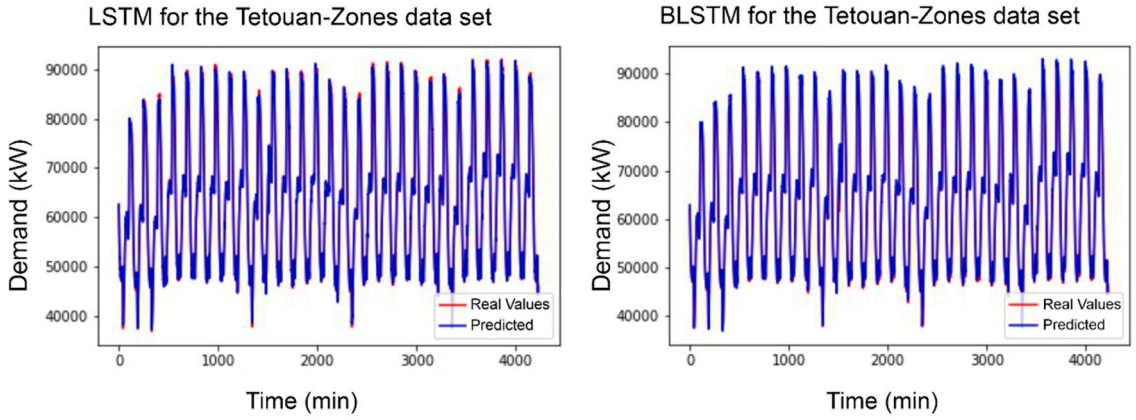


Fig. 11. Typical LSTM and BLSTM predictions for the holdout set (Tetouan-Zones data set).

Table 10
LSTM models' results for the Singapore holdout subset.

LSTM Model	Metrics - LSTM models - Singapore holdout				
	RMSE	NRMSE	MAE	MAPE (%)	R^2
1	35.096	0.016	28.374	0.6	99.7
2	35.797	0.017	28.957	0.6	99.7
3	30.836	0.014	24.218	0.5	99.8
4	42.846	0.020	35.388	0.7	99.6
5	32.755	0.015	26.533	0.5	99.7
6	48.184	0.022	39.409	0.8	99.5
7	31.026	0.014	24.818	0.5	99.8
8	46.894	0.022	40.107	0.8	99.5
9	36.283	0.017	28.940	0.6	99.7
10	29.291	0.014	23.562	0.5	99.8
Average	36.901	0.017	30.031	0.6	99.7
Median	35.446	0.017	28.657	0.6	99.7
St-Dev	6.781	0.003	6.131	0.1	0.1
Min	29.291	0.014	23.562	0.5	99.5
Max	48.184	0.022	40.107	0.8	99.8

Table 11
BLSTM models' results for the Singapore holdout subset.

BLSTM Model	Metrics - BLSTM models - Singapore holdout				
	RMSE	NRMSE	MAE	MAPE (%)	R^2
1	39.755	0.019	30.713	0.6	99.6
2	74.322	0.035	63.770	1.2	98.7
3	27.914	0.013	22.190	0.5	99.8
4	29.148	0.014	22.167	0.4	99.8
5	33.638	0.016	26.725	0.5	99.7
6	24.266	0.011	18.874	0.4	99.9
7	29.963	0.014	25.081	0.5	99.8
8	22.716	0.011	17.727	0.4	99.9
9	24.229	0.011	18.644	0.4	99.9
10	29.269	0.014	23.618	0.5	99.8
Average	33.522	0.016	26.951	0.5	99.7
Median	29.209	0.014	22.904	0.5	99.8
St-Dev	15.179	0.007	13.537	0.3	0.4
Min	22.716	0.011	17.727	0.4	98.7
Max	74.322	0.035	63.770	1.2	99.9

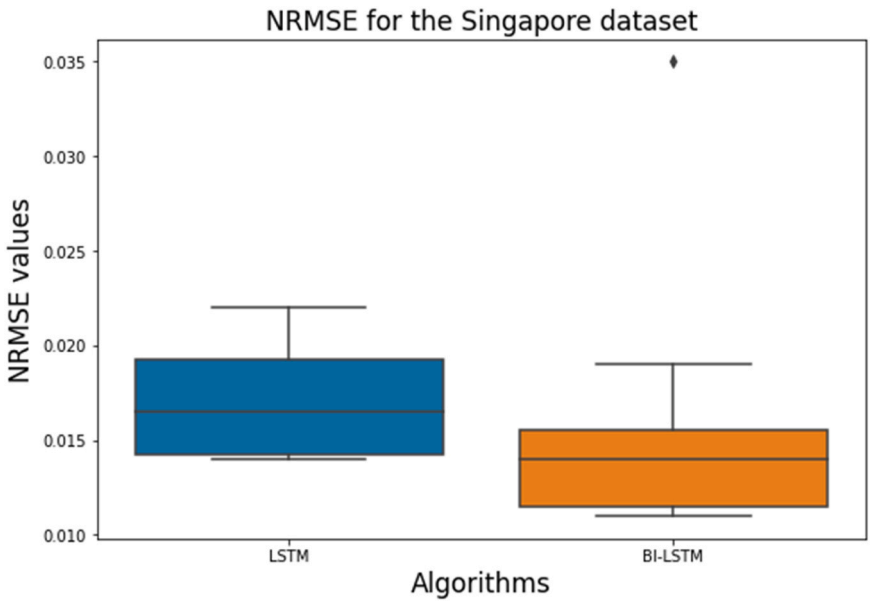


Fig. 12. NRMSE boxplots for LSTM and BLSTM models applied to the Singapore holdout subset.

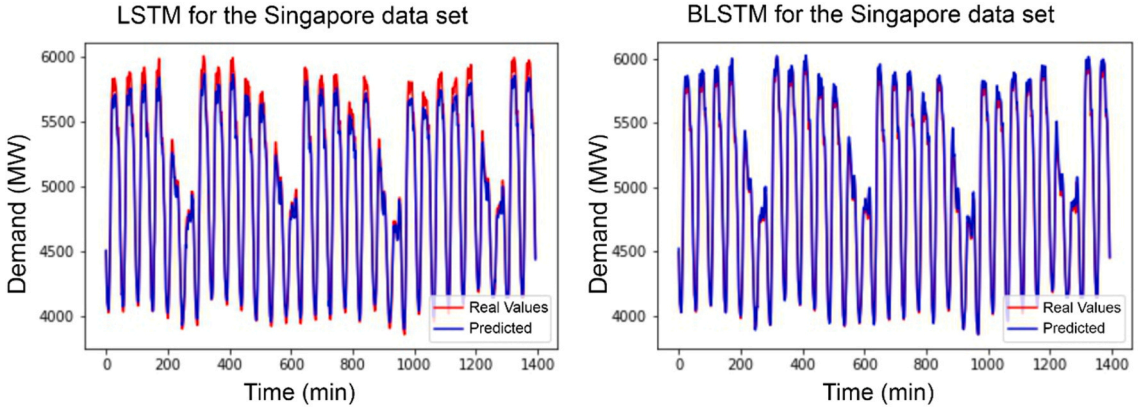


Fig. 13. Typical LSTM and BLSTM predictions for the holdout set (Singapore data set).

Table 12
Table with consolidated results (average RMSE and R^2) for all data sets. Best NRMSE averages in bold.

Network	UCI-Household		LABIC-Building		Tetouan-Zones		Singapore	
	NRMSE	R^2	NRMSE	R^2	NRMSE	R^2	NRMSE	R^2
LSTM	0.070	69.5	0.032	97.8	0.016	99.6	0.017	99.7
BLSTM	0.067	72.6	0.031	97.9	0.012	99.8	0.016	99.7

Table 13
LSTM and BLSTM's execution times (in hours) for the TS-CV of each data set (k = 10).

Networks	UCI-Household	LABIC-Building	Tetouan-Zones	Singapore
	Time (h)	Time (h)	Time (h)	Time (h)
LSTM	8.75	1.58	2.28	0.97
BLSTM	17.76	2.58	3.93	1.63

consumption data sets could also be incorporated for such an investigation. Additionally, Transfer Learning (Otović et al., 2022; Ahn and Kim, 2022) could be investigated according to the methodology presented.

Multivariate energy consumption TS was not included in the present study, which may be considered a limitation. Other tests may be performed considering the influence of environmental variables such as temperature and solar irradiance for improving the performance of the predictions. However, for univariate TS, the consistent lower NRMSE and therefore better performance due to the bidirectional information flow of BLSTM needs to be considered for choosing a prediction model for energy consumption TS.

7. Conclusion

Currently, energy efficiency is a central problem both from economic and environmental perspectives. New technologies must be tested for monitoring and predicting electric energy consumption, aiming to

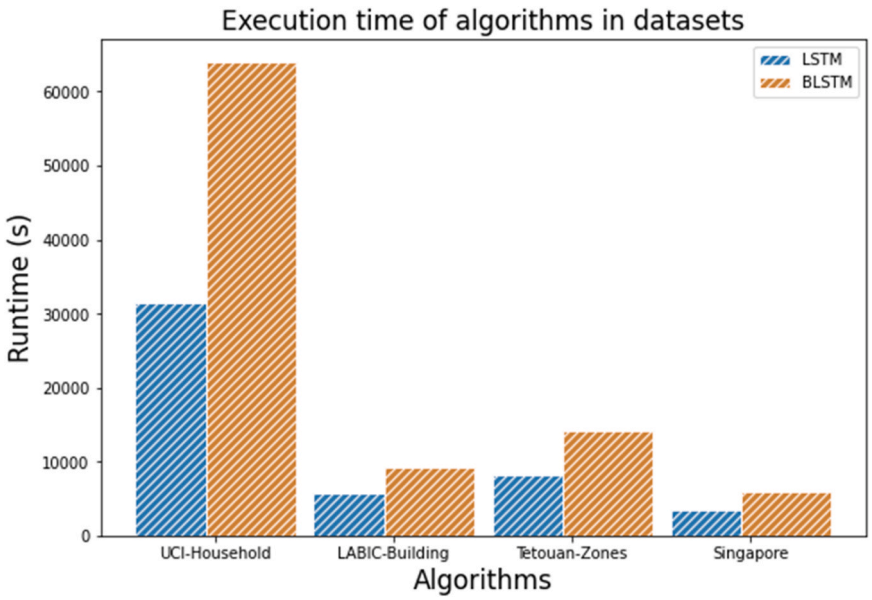


Fig. 14. LSTM and BLSTM's execution times for all data sets.

reduce it through concrete actions, as highlighted by Gardner and Stern (2002).

Thus, a comparison between DNNs for electric energy consumption time series prediction becomes important in such scenarios. LSTM and BLSTM are prominent DNN architectures that were compared in the present study. The statistical comparison of models generated with different lengths of points (generated by TS-CV) and applied to holdout sets is a way of demonstrating their robustness, also taking into account that the TSs were selected considering different scales and characteristics (household, building, city zones and country scales).

As a result, BLSTM outperformed LSTM models for energy consumption time series prediction with statistical significance, despite the higher time necessary for training, which compensates for the better results.

Finally, it is important to consider important points such as performance (high values), robustness (consistency of results) and training time (guarantees good results, despite the longer training time, but statistically presents better results).

Thus, the present study provides statistical evidence supporting that BLSTM outperforms LSTM models according to the tests performed, based on a complete methodology for TS prediction and also establishes a baseline for future investigation of electric energy consumption TS prediction.

CRediT authorship contribution statement

Davi Guimarães da Silva: Conceptualization, Methodology, Software, Visualization, Data curation, Validation, Writing – original paper. **Anderson Alvarenga de Moura Meneses:** Conceptualization, Methodology, Validation, Analysis, Writing – review & editing.

Declaration of Competing Interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Davi Guimarães da Silva reports financial support was provided by Brazilian National Council for Scientific and Technological Development (CNPq).

Data availability

Data will be made available on request.

Acknowledgments

D.G.S. and A.A.M.M. acknowledge the Brazilian National Council for Scientific and Technological Development (CNPq) for financial support. The authors would like to thank Dr. Troy Beldini for proofreading the article and reviewers for their valuable comments and suggestions.

References

Ahmad, T., et al., 2022. Energetics systems and artificial intelligence: applications of industry 4.0. Energy Rep. 8, 334–361. https://doi.org/10.1016/j.egyr.2021.11.256.

Ahn, Y., Kim, B.S., 2022. Prediction of building power consumption using transfer learning-based reference building and simulation dataset. Energy Build. 258, 111717 https://doi.org/10.1109/TSG.2019.2938068.

Baldi, Pierre, et al., 1999. Exploiting the past and the future in protein secondary structure prediction. Bioinformatics 15 (11), 937–946. https://doi.org/10.1093/bioinformatics/15.11.937.

Barrow, D., et al., 2013. Ranking rankings: an empirical comparison of the predictive power of sports ranking methods. J. Quant. Anal. Sports 9 (2), 187–202. https://doi.org/10.1515/jqas-2013-0013.

Bianchi, F.M., et al., 2020. Reservoir computing approaches for representation and classification of multivariate time series. IEEE Trans. Neural Netw. Learn. Syst. 32 (5), 2169–2179. https://doi.org/10.1109/TNNLS.2020.3001377.

Bohara, B., et al., 2022. Short-term aggregated residential load forecasting using BiLSTM and CNN-BiLSTM. In: Proceedings of the International Conference on Innovation and Intelligence for Informatics, Computing, and Technologies (3ICT), IEEE, pp. 37–43. ⟨https://doi.org/10.1109/3ICT56508.2022.9990696⟩.

Bunn, D.W., 2000. Forecasting loads and prices in competitive power markets. Proc. IEEE 88 (2), 163–169. https://doi.org/10.1109/5.823996.

Caro, E., Juan, J., Cara, J., 2020. Periodically correlated models for short-term electricity load forecasting. Appl. Math. Comput. 364, 124642 https://doi.org/10.1016/j.amc.2019.124642.

Chen, J., Chaudhari, N.S., 2005. Protein secondary structure prediction with a hybrid RNN/HMM system. In: Proceedings of the IEEE International Joint Conference on Neural Networks, IEEE, pp. 538–541. ⟨https://doi.org/10.1109/IJCNN.2005.1555888⟩.

Cheng, Y.L., Lim, M.H., Hui, K.H., 2022. Impact of internet of things paradigm towards energy consumption prediction: a systematic literature review. Sustain. Cities Soc. 78, 103624 https://doi.org/10.1016/j.scs.2021.103624.

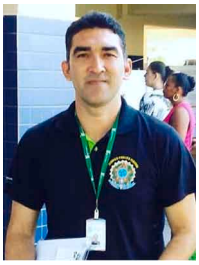
Chollet, F., 2018. Deep Learning with Python. Shelter Island, NY, USA.

Da Silva, D.G., Geller, M.T.B., Moura, M.S.S., Meneses, A.A.M., 2022. Performance evaluation of LSTM neural networks for consumption prediction. E-Prime Adv. Electr. Eng. Electron. Energy 2, 100030. https://doi.org/10.1016/j.prime.2022.100030.

Da Silva, D.G., Geller, M.T.B., Moura, M.S.S., Meneses, A.A.M., 2022. Performance evaluation of LSTM neural networks for consumption prediction. E-Prime Adv.

Electr. Eng. Electron. Energy 2, 100030. https://doi.org/10.1016/j.prime.2022.100030 (Available online: ⟨https://www.kaggle.com/datasets/daviguimaraes/labic-building-data-set⟩).
Da Silva, D.G., Geller, M.T.B., Moura, M.S.S., Meneses, A.A.M., 2021. A deep learning prediction module for the IoT system energysaver for monitoring and estimating power consumption. In: Proceedings of the 16th Conference on Sustainable Development of Energy, Water and Environment Systems (SDEWES), Dubrovnik, Croatia.
Das, A., et al., 2020. Occupant-centric miscellaneous electric loads prediction in buildings using state-of-the-art deep learning methods. Appl. Energy 269, 115135. https://doi.org/10.1016/j.apenergy.2020.115135.
Demšar, J., 2006. Statistical comparisons of classifiers over multiple data sets. J. Mach. Learn. Res. 7, 1–30.
Dietterich, T., 2002. Machine learning for sequential data: a review. Struct. Syntactic Stat. Pattern Recognit. 15–30. https://doi.org/10.1007/3-540-70659-3_2 https://doi.org/10.1007/3-540-70659-3_2.
Energy Market Company Pte Ltd (EMC), 2010. Uniform Singapore Energy Price and DemandForecast. Available at: ⟨https://www.emcsg.com/marketdata/priceinformation⟩ (Accessed 22 Jun 2022). Described in this manuscript as Singapore Data Set.
Fang, X., Yuan, Z., 2019. Performance enhancing techniques for deep learning models in time series forecasting. Eng. Appl. Artif. Intell. 85, 533–542. https://doi.org/10.1016/j.engappai.2019.07.011.
Farsi, B., et al., 2021. On short-term load forecasting using machine learning techniques and a novel parallel deep LSTM-CNN approach. IEEE Access 9, 31191–31212. https://doi.org/10.1109/ACCESS.2021.3060290.
Fernández-Martínez, D., Jaramillo-Morán, M.A., 2022. Multi-step hourly power consumption forecasting in a healthcare building with recurrent neural networks and empirical mode decomposition. Sensors 22, 3664. https://doi.org/10.3390/s22103664.
FIESC – Federation of Industries of Santa Catarina State, 2022. Electrical Waste in Brazil Is Equivalent to the Consumption of 20 Million Homes. Available at: ⟨https://fiesc.com.br/pt-br/imprensa/desperdicio-eletrico-no-brasil-equivale-ao-consumo-de-20-milhoes-de-residencias⟩ (Accessed on Dec. 2022; in Portuguese).
Friedman, M., 1937. The use of ranks to avoid the assumption of normality implicit in the analysis of variance. J. Am. Stat. Assoc. 32 (200), 675–701. https://doi.org/10.1080/01621459.1937.10503522.
Fukada, T., Schuster, M., Sagisaka, Y., 1999. Phoneme boundary estimation using bidirectional recurrent neural networks and its applications. Syst. Comput. Jpn. 30 (4), 20–30. https://doi.org/10.1002/(SICI)1520-684X(199904)30:4%3C20::AID-SCJ3%3E3.0.CO;2-E.
García, S., et al., 2010. Advanced nonparametric tests for multiple comparisons in design of experiments in computational intelligence and data mining: experimental analysis of power. Inf. Sci. 180, 2044–2064. https://doi.org/10.1016/j.ins.2009.12.010.
Gardner, G.T., Stern, P.C., 2002. Environmental Problems and Human Behavior, second ed. Pearson Custom Publishing, Boston, MA.
Gers, F.A., Schmidhuber, E., 2001. LSTM recurrent networks learn simple context-free and context-sensitive languages. IEEE Trans. Neural Netw. 12 (6), 1333–1340. https://doi.org/10.1109/72.963769.
Gottam, S., et al., 2021. A CNN-LSTM model trained with grey wolf optimizer for prediction of household power consumption. In: Proceedings of the IEEE International Symposium on Smart Electronic Systems (iSES), IEEE, pp. 355–360. ⟨https://doi.org/10.1109/iSES52644.2021.00089⟩.
Graves, A., et al., 2008. A novel connectionist system for unconstrained handwriting recognition. IEEE Trans. Pattern Anal. Mach. Intell. 31 (5), 855–868. https://doi.org/10.1109/TPAMI.2008.137.
Graves, A., Schmidhuber, J., 2005. Framewise phoneme classification with bidirectional LSTM and other neural network architectures. Neural Netw. 18 (5–6), 602–610. https://doi.org/10.1016/j.neunet.2005.06.042.
Graves, A., Schmidhuber, J., 2008. Offline handwriting recognition with multidimensional recurrent neural networks. Advances in Neural Information Processing Systems 21. MIT Press, Cambridge, MA, pp. 545–552.
Graves, A., 2012. Long short-term memory. Supervised Sequence Labelling with Recurrent Neural Networks, pp. 37–45. ⟨https://doi.org/10.1007/978-3-642-24797-2_4⟩.
Hadri, S., Najib, M., Bakhouya, M., Fakhri, Y., El Arroussi, M., 2021. Performance evaluation of forecasting strategies for electricity consumption in buildings. Energies 14, 5831. https://doi.org/10.3390/en14185831.
Hasan, M.N., et al., 2019. Electricity theft detection in smart grid systems: a CNN-LSTM based approach. Energies 12 (17), 3310. https://doi.org/10.3390/en12173310.
Haykin, S., 2009. Neural Networks and Learning Machines, third ed. Pearson Education India.
Hebrail, G., Berard, A., 2012. Individual household electric power consumption data set. UCI Machine Learning Repository. Available online: ⟨https://archive.ics.uci.edu/dataset/235/individual+household+electric+power+consumption⟩ (Accessed 22 August 2023). Described in this manuscript as UCI-Household Data Set.
Hewamalage, H., Ackermann, K., Bergmeir, C., 2023. Forecast evaluation for data scientists: common pitfalls and best practices. Data Min. Knowl. Discov. 37 (2), 788–832. https://doi.org/10.1007/s10618-022-00894-5.
Himeur, Y., et al., 2021. Artificial intelligence based anomaly detection of energy consumption in buildings: a review, current trends and new perspectives. Appl. Energy 287, 116601. https://doi.org/10.1016/j.apenergy.2021.116601.
Hochreiter, S., Schmidhuber, J., 1997. Long short-term memory. Neural Comput. 9 (8), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735.
Hou, T., et al., 2021. A novel short-term residential electric load forecasting method based on adaptive load aggregation and deep learning algorithms. Energies 14 (22). https://doi.org/10.3390/en14227820.
Hyndman, R., Athanasopoulos, G., 2018. Forecasting: Principles and Practice, third ed. OTexts, Melbourne, Australia. OTexts.com/fpp3.
Jogunola, O., et al., 2021. Comparative analysis of hybrid deep learning frameworks for energy forecasting. In: Proceedings of the 5th International Conference on Future Networks & Distributed Systems, pp. 214–219. ⟨https://doi.org/10.1145/3508072.3508105⟩.
Jurafsky, D., Martin, J.H., 2000. Speech and Language Processing: an Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition. Prentice Hall Prentice Hall.
Kaur, D., et al., 2019. Smart grid energy management using RNN-LSTM: a deep learning-based approach. In: Proceedings of the IEEE Global Communications Conference (GLOBECOM), IEEE, pp. 9–13. ⟨https://doi.org/10.1109/GLOBECOM38437.2019.9013850⟩.
Khan, N., et al., 2021. DB-Net: a novel dilated CNN based multi-step forecasting model for power consumption in integrated local energy systems. Int. J. Electr. Power Energy Syst. 133, 107023 https://doi.org/10.1016/j.ijepes.2021.107023.
Kim, T.-Y., Cho, S.-B., 2018. Predicting the household power consumption using CNN-LSTM hybrid networks. Intelligent Data Engineering and Automated Learning–IDEAL 2018: 19th International Conference, Madrid, Spain, November 21–23, 2018, Proceedings, Part I 19. Springer International Publishing, pp. 481–490. https://doi.org/10.1007/978-3-030-03493-1_50.
Kitchenham, Barbara, et al., 2009. Systematic literature reviews in software engineering–a systematic literature review. Inf. Softw. Technol. 51 (1), 7–15. https://doi.org/10.1016/j.infsof.2008.09.009.
Le, T., et al., 2019. Improving electric energy consumption prediction using CNN and Bi-LSTM. Appl. Sci. 9 (20), 4237. https://doi.org/10.3390/app9204237.
Lee, S.H., et al., 2019. Energy consumption prediction system based on deep learning with edge computing. In: Proceedings of the IEEE 2nd International Conference on Electronics Technology (ICET), IEEE, pp. 473–477. ⟨https://doi.org/10.1109/ELTECH.2019.8839589⟩.
Liang, Y., Deng, J., Cui, B., 2020. Bidirectional LSTM: an innovative approach for phishing URL identification. Innovative Mobile and Internet Services in Ubiquitous Computing: Proceedings of the 13th International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS-2019). Springer International Publishing, pp. 326–337. https://doi.org/10.1007/978-3-030-22263-5_32.
Martinez, D., Ebenhack, B.W., Wagner, T., 2019. Energy Efficiency: Concepts and Calculations. Elsevier. https://doi.org/10.1016/C2016-0-02161-7.
Mellouli, N., Akerma, M., Hoang, M., Leducq, D., Delahaye, A., 2019. Deep learning models for time series forecasting of indoor temperature and energy consumption in a cold room. In: Nguyen, N., Chbeir, R., Exposito, E., Aniorté, P., Trawiński, B. (Eds.), Computational Collective Intelligence - Part II. ICCCI 2019. Lecture Notes in Computer Science, vol. 11684. Springer, Cham. https://doi.org/10.1007/978-3-030-28374-2_12.
Moon, J., et al., 2019. Temporal data classification and forecasting using a memristor-based reservoir computing system. Nat. Electron. 2 (10), 480–487. https://doi.org/10.1038/s41928-019-0313-3.
Mozer, M.C., 1993. Neural net architectures for temporal sequence processing. Santa Fe Institute Studies in the Sciences of Complexity-Proceedings Volume. Addison-Wesley Publishing Co, p. 243-243.
Mubashar, R., et al., 2022. Efficient residential load forecasting using deep learning approach. Int. J. Comput. Appl. Technol. 68 (3), 205–214. https://doi.org/10.1504/IJCAT.2022.124940.
Nemenyi, P.B., 1963. Distribution-Free Multiple Comparisons (Ph.D. thesis). Princeton University.
Otović, E., et al., 2022. Intra-domain and cross-domain transfer learning for time series data—how transferable are the features? Knowl. Based Syst. 239, 107976 https://doi.org/10.1016/j.knosys.2021.107976.
Ozer, I., Efe, S.B., Ozbay, H., 2021. A combined deep learning application for short term load forecasting. Alex. Eng. J. 60 (4), 3807–3818. https://doi.org/10.1016/j.aej.2021.02.050.
Rafi, S.H., Al-Masood, N., Deeba, S.R., Hossain, E., 2021. A short-term load forecasting method using integrated CNN and LSTM network. IEEE Access 9, 32436–32448. https://doi.org/10.1109/ACCESS.2021.3060654.
Reimers, N., Gurevych, I., 2017. Optimal Hyperparameters for Deep Lstm-networks for Sequence Labeling Tasks. ArXiv preprint arXiv:1707.06799. ⟨https://doi.org/10.48550/arXiv.1707.06799⟩.
Rhif, M., et al., 2020. Deep learning models performance for NDVI time series prediction: a case study on north west Tunisia. In: Proceedings of the Mediterranean and Middle-East Geoscience and Remote Sensing Symposium (M2GARSS), IEEE, pp. 9–12. ⟨https://doi.org/10.1109/M2GARSS47143.2020.9105149⟩.
Robinson, A.J., Fallside, F., 1987. The Utility Driven Dynamic Error Propagation Network. Technical Report CUED/F-INFENG/TR.1. Cambridge University Engineering Department.
Rumelhart, D.E., Hinton, G.E., Williams, R.J., 1986. Learning representations by back-propagating errors. Nature 323 (6088), 533–536. https://doi.org/10.1038/323533a0.
Salam, A., El Hibaoui, A., 2018. Comparison of machine learning algorithms for the power consumption prediction:-case study of tetouan city. In: Proceedings of the 6th International Renewable and Sustainable Energy Conference (IRSEC), IEEE, pp. 1–5. Available at: ⟨https://archive.ics.uci.edu/ml/datasets/Power+consumption+of+Tetouan+city⟩ (Accessed 22 Jun 2022). Described in this manuscript as Tetouan-Zones Data Set.
Schirmer, P.A., Mporas, I., Potamitis, I., 2019. Evaluation of regression algorithms in residential energy consumption prediction. In: Proceedings of the 3rd European Conference on Electrical Engineering and Computer Science (EECS), Athens, Greece, 28–30, December, pp. 22–25. ⟨https://doi.org/10.1109/EECS49779.2019.00018⟩.

Schmidhuber, J., 1992. A fixed size storage O (n^3) time complexity learning algorithm for fully recurrent continually running networks. Neural Comput. 4 (2), 243–248. https://doi.org/10.1162/neco.1992.4.2.243.
Schmidhuber, J., 2015. Deep learning in neural networks: an overview. Neural Netw. 61, 85–117. https://doi.org/10.1016/j.neunet.2014.09.003.
Schuster, M., Paliwal, K.K., 1997. Bidirectional recurrent neural networks (Nov). IEEE Trans. Signal Process. 45 (11), 2673–2681. https://doi.org/10.1109/78.650093. ⟨https://doi.org/10.1109/78.650093⟩.
Serpanos, D., Wolf, M., 2018. Internet-of-Things (IoT) Systems. Springer International Publishing, Cham. https://doi.org/10.1007/978-3-319-69715-4_5.
Shao, X., et al., 2020. Domain fusion CNN-LSTM for short-term power consumption forecasting. IEEE Access 8, 188352–188362. https://doi.org/10.1109/ACCESS.2020.3031958.
Shaqour, A., et al., 2022. Electrical demand aggregation effects on the performance of deep learning-based short-term load forecasting of a residential building. Energy AI 8, 100141. https://doi.org/10.1016/j.egyai.2022.100141.
Sharfuddin, A.A., Tihami, M.N., Islam, M.S., 2018. A deep recurrent neural network with BLSTM model for sentiment classification. In: Proceedings of the International Conference on Bangla Speech and Language Processing (ICBSLP), IEEE, pp. 1–4. ⟨https://doi.org/10.1109/ICBSLP.2018.8554396⟩.
Shin, S.-Y., Woo, H.-G., 2022. Energy consumption forecasting in korea using machine learning algorithms. Energies 15 (13), 4880. https://doi.org/10.3390/en15134880.
Siami-Namini, S., Tavakoli, N., Namin, A.S., 2019. The performance of LSTM and BLSTM in forecasting time series. In: Proceedings of the IEEE International Conference on Big Data (Big Data), IEEE, pp. 3285–3292. ⟨https://doi.org/10.1109/BigData47090.2019.9005997⟩.
Singh, A.P., et al., 2018. Tetuan City Power Consumption. Distribution Network Station of Tetouan city in Morocco. ⟨https://www.kaggle.com/datasets/gmkeshav/tetuan-city-power-consumption⟩ (Accessed 22 Jun 2022).
Witten, I.H., Frank, E., 2002. Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations. Association for Computing Machinery, New York, USA, pp. 76–77. https://doi.org/10.1145/507338.507355.
Zhao, Y., et al., 2018. Applying deep bidirectional LSTM and mixture density network for basketball trajectory prediction. Optik 158, 266–272. https://doi.org/10.1016/j.ijleo.2017.12.038.



Davi Guimarães da Silva Master in Informatics from the Federal University of Amazonas (UFAM - 2016) and Doctorate in Environmental Sciences from the Federal University of Western Pará (UFOPA-2023). Specialization in Teaching for Professional Education (2018). Specialization in Systems Engineering (2013). MBA Specialization in Environment (2011). Graduated in Technology in Data Processing from the Federal University of Pará (UFPA - 2009). He is currently a Professor at the Federal Institute of Education, Science and Technology of Pará (IFPA). He is interested in research in the areas of: Database and Information Retrieval; Machine Learning; Deep Learning (Time Series prediction and Digital Image Processing, with Deep Neural Networks); Data Mining; Information Technologies applied to Education; Virtual Learning Objects; Augmented Reality Applied to Education.

Anderson Alvarenga de Moura Meneses Graduated in Physics from the Federal University of Rio de Janeiro (2000), in Brazil. Master (2005) and doctorate (2010) degrees in Nuclear Engineering from COPPE Institute, at the Federal University of Rio de Janeiro, with a fellowship at the Dalle Molle Institute for Artificial Intelligence (IDSIA, University of Lugano, Switzerland) in 2009. Specialist in Systems Analysis, Design and Management from Pontifical Catholic University, in Rio de Janeiro (2008). Associate Professor at the Federal University of Western Pará (UFOPA). Head of the Computational Intelligence Laboratory (LabIC/UFOPA) since 2015. Leader of the Computational Intelligence and Optimization research group. Permanent member of the Postgraduate Program in Amazon Natural Resources (PPGRNA/UFOPA). Collaborating member of the Postgraduate Program in Society, Nature and Development (PPGSND/UFOPA). Researcher PQ2 granted by the Brazilian National Council for Scientific and Technological Development (CNPq/Brazil). Areas of interest: Nuclear Engineering (optimization of nuclear reactor fuel reload), Deep Learning (Time Series Prediction and Image Processing) and Energy (Artificial Intelligence applied to electrical energy consumption monitoring).