Received: March 15, 2022. Revised: April 12, 2022.

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

A Novel LSTM-GRU-Based Hybrid Approach for Electrical Products Demand Forecasting

Aicha El filali[1]* **Aissam Jadli[2]** **El Habib Ben Lahmer[1]** **Sanaa El filali[1]**

[1]*Faculty of Sciences Ben M'sick, Hassan II University, Casablanca, Morocco*
[2]*AICSE Laboratory, ENSAM Casablanca, University Hassan II, Casablanca, Morocco*
* Corresponding author's Email: aicha.elfilali-etu@etu.univh2c.ma

Abstract: Demand forecasting is an indispensable key to planning and achieving objectives, as it feeds all the processes in the company's supply chain. Forecasting is a difficult task that requires a lot of analysis using powerful complex mathematical equations to build a model capable of predicting customer behavior according to the market variability involved. Therefore, the objective of this paper is to build a new hybrid deep learning method named (LSTM-GRU) based on sequential learning using long short term memory (LSTM) and gated recurrent units (GRU). The proposed method builds automatically the best prediction model by considering various combinations of LSTM and GRU hyperparameters using the gridsearch technique. It can predict highly fluctuating demand data while taking into consideration the non-linear characteristics of time series. The new method's performance was compared to several deep learning models, namely single layer LSTM, single layer GRU, stacked LSTM, and stacked GRU, using real electrical product data from an industrial Moroccan company. The evaluation of the experimental results with mean absolute error (MAE) and root mean squared error (RMSE) methods reveal that the predictions produced with the new model are the most accurate with the lowest error rates MAE= 237682.56 and RMSE= 366633.28 respectively.

Keywords: Demand forecasting, Supply chain, Deep learning, Long short term memory, Gated recurrent units, Gridsearch technique, Mean absolute error, Mean squared error.

1. Introduction

Demand forecasting is a sensitive task for the company as many activities are derived from it. According to [1], demand forecasting is the starting point for all activities within the supply chain (SCM). The supply chain (SCM) encompasses the management of a company's processes from planning to distribution, it aims to implement an optimal management of their processes in maintaining a supply that matches the demand, it is considered the first to be affected by the quality of the forecasts [2, 3]. Obtaining forecasts is a complicated process where each phase must be carried out with great care, involving, in particular, the stage of data recovery and analysis, the key point for producing good forecasts. The overuse of technology combined with the Internet of Things (IIoT) and mobile internet have produced an excessive number of massive data, called big data. Big data is a collection of data defined by the five Vs: volume, velocity, variety, veracity, and value [4-6]. The study [5] has demonstrated that the availability of big data in the supply chain has caused a remarkable complexity in the demand forecasting operation. In addition, companies have moved towards sourcing products from distant markets around the world to save production costs [7]. This, in turn, has caused an extension of both production and product supply lead times. According to [8, 9], as the forecasting horizon gets longer, the accuracy of demand forecasting also tends to decrease. To this end, the quest for a reliable forecast of customer demand is one of the most dynamic problems in supply chain management [10]. Statistical forecasting methods such as moving average, exponential smoothing, Box-Jenkins methods, etc., have proven unable to analyze and capture the characteristics of these new and potentially invaluable data sets to produce good forecasts. For this reason, companies can get more out of machine learning to overcome the limitations of statistical approaches and produce more accurate demand forecasts, through its ability to detect and learn non-linear relationships in large masses of raw data. The electrical manufacturing industry has been facing great challenges in recent years: manufacturing products that contribute to safety, energy efficiency, longer life, as well as connected objects, as well as optimizing product prices. The response to these challenges requires companies in this field to follow the new technology evolution, to meet the customer needs. Considering that the customer demand is modelled in most of the time in the form of sequential data [11], then demand forecasting can be considered as a time series forecasting problem [11, 12]. A time series is a sequence of observed data over time [13]. ANNs are among the ML models frequently used for time series forecasting [11, 14], this is due to several reasons, firstly, ANNs are one of the self-parametric methods based on data, which give them the ability to learn through experience and capture the existing unknown or hard to describe relationships between data [15], secondly, ANNs can generalize. After the presentation of the data (a sample), ANNs can at the existence of noisy information, deduce the future behavior of the unobserved part of the population from the sample of the past behavior. Finally, ANNs are non-linear models, and real-world data are often non-linear so ANNs are very appropriate for this environment. Recurrent neural networks (RNN) are one of the variants of ANNs, they use feedback loops to allow signals moving in different directions between nodes [16]. RNNs are dedicated to learning sequential type data to predict the next most likely scenario. Although RNNs can capture time-series non-linearity in many areas of the supply chain [17], the literature has confirmed that they are unable to produce accurate predictions in the case of heterogeneous data with a high degree of non-linearity [18, 19], this is due to their short memory and the problem of gradient vanishing, thus leading to a learning difficulty [20]. In this sense, the deep learning architectures long short term memory (LSTM) [21] and gated recurrent units (GRU) [13] are designed to overcome these limitations. Research in the supply chain domain has strived to have better performing customer demand forecasts by integrating LSTM [19, 22, 23], and GRU methods [24, 25].

To this end, we have proposed in this research work a new hybrid deep learning model dedicated to the electric products demand forecasting in the Moroccan industrial context called (LSTM-GRU), we have obtained a state-of-the-art accuracy by applying one of the optimization techniques namely the gridsearch method. The new model's performance is evaluated on real data presenting the sales turnover history of the product that makes the largest turnover for the company INGELEC. There are many publications on LSTM and GRU applications but to our knowledge, none of them have been applied in electrical manufacturing companies. To confirm the superiority of the proposed hybrid model (LSTM-GRU), we compared the forecasting results with those of the simple LSTM, simple GRU, stacked LSTM, and stacked GRU methods using the MAE and RMSE evaluation measures. The main contributions of this paper are summarized as follows:

1. The actual sales history used in this research, is a time series for the last five years of an electrical product sold by a Moroccan company. We began by presenting it and applying statistical analysis of the series characteristics, then we pre-processed the data so that they are ready for modelling.
2. The design process of the proposed model (LSTM-GRU) based on the gridsearch method is described step by step and the experimental results on real data are validated and presented at the end of this paper.
3. The efficiency of the proposed new hybrid model (LSTM-GRU) is compared with simple LSTM, simple GRU, stacked LSTM, and stacked GRU models and the results show the superiority of the proposed model in terms of accuracy.
4. The prediction results obtained are shown to be consistent with the expectations of related works obtained using the proposed method.

The rest of this work is structured in four sections. In the second section, we present a literature review of applications of deep learning methods to time series forecasting in the industrial domain. In the third section, we describe the main characteristics of the dataset used in this research and then the methodology followed for the elaboration of the new hybrid LSTM-GRU model proposed for this application. In the fourth section, we present the study's experimental description. In the fifth section, we compare the performance of the new LSTM-GRU model with simple LSTM, simple GRU, stacked LSM, and stacked GRU. Finally, in the sixth section, we end with a conclusion and suggestions for future work.

Received: March 15, 2022. Revised: April 12, 2022.

2. Related work

In the industrial field of electrical products manufacturing, the fluctuation of supply, the wide variety of products, and the uncertainty of demand is increasing more and more, a correct forecast of customer demand can be a challenge in this environment.

Traditionally, demand forecasting has been performed by statistical methods including linear models such as moving average, weighted average, multivariate linear regression model [26].

These methods have shown advantages in modelling customer demand, however, it is very difficult to consider that demand is linear in the complex environment of industrial companies. Some researchers have suggested that demand is a series of time series data, they have applied the linear models: moving average, autoregressive moving average, ARIMA, and SARIMA for demand forecasting. These methods are simple and fast to implement, however, the predicted results are unstable which confirms that they do not adapt to real demand characteristic changes in the industrial world. Then, several non-linear methods have been developed to make the forecast more compatible with the customer demand, such as ARCH (autoregressive conditional heteroscedastic), GARCH (general autoregressive conditional heteroscedastic) [27]. Nevertheless, each of them can only model one nonlinearity, which makes the forecasting operation even more complex. Recently, ML techniques including support vector machines (SVM), artificial neural networks (ANN) [28], and Bayesian networks, have been often applied to solve the time series prediction problem [11].

The systemic literature on the applications of ML techniques suggests that during the last three years ANNs and their variants are the most used methods for demand forecasting in the industrial domain [29], [30]. This is due to their universal approximation and their ability to capture the nonlinearity of data. Recurrent neural networks (RNN) differ from traditional ANNs concerning parameter selection. RNNs share the same training parameters which red their numbers considerably, unlike ANNs which select different parameters in each layer [31]. However, RNNs still present the problem of vanishing gradient.

The LSTM and GRU methods are two variants of deep RNNs that have recently attracted more attention in the field of time series forecasting, however, there are still a modest number of real applications in the industrial field. The LSTM can store time dependence information for a long period allowing it to forecast long-term time-series data accurately. This performance has been confirmed in several research works, as explained by the authors in [24], deep LSTM was applied in the industry 4.0 setting on real data with six years' history of five products. The authors compared LSTM results using root mean square deviation (RMSE) with statical methods SARIMAX and Triple exponential smoothing (ETS) and ML methods random forest (RF), extreme gradient boosting (XGBoost), and single-layer LSTM. The experiment showed that the lowest error values were obtained with the ETS, XGBoost, LSTM, and MLP models for each product, but when measuring the overall score of each model, the deep LSTM obtained the best with a value of 81.9 %. Similarly, for [32], in the oil industry, the researchers proposed a deep LSTM architecture named by (DLSTM) for time series forecasting of two oil fields, the performance of the new model was compared with ARIMA, deep gated recurrent unit (DGRU), RNN, deep LSTM, nonlinear extension for linear arps decline (NEA), and higher-order neural network (HONN) models. For the two oil fields cases, the results showed that DLSTM can produce the most accurate forecasts with the lowest errors RMSE = 0.209, RMSPE= 2.995 for the first oil field, and RMSE= 0.025 and RMSPE= 3.496 in the case of the second field. GRU method is a specific deep learning model based on LSTM structure, it converges quickly and has fewer parameters than LSTM with similar accuracy. The applications of GRU in the industrial domain are very rare, we can mention, [33] where GRU was applied for residential energy demand forecasting, in this research, three GRU models were built respectively with one hidden layer (GRU - M28), two layers (GRU - M2,20) and three hidden layers (GRU - M3,15) of ten to thirty nodes by changing the training parameters. Using mean absolute percentage error (MAPE), the experiment showed that the model (GRU - M2,20) parameterized with two hidden layers with twenty nodes each, outperformed the one and three hidden layer models in producing the forecasts with the lowest errors, MAPE= 0.79. In the same context, the authors in [34] proposed a method to forecast the load of a residential community with the GRU model. To analyze the influencing factors on the residential load, Lasso regression and partial correlation analysis are used. The proposed model has been improved with the MISO structure and the ADAM algorithm. The experiment showed that GRU at different season scales is superior compared to traditional LSTM and RNN in terms of convergence speed, higher performance with less error MAPE= 6.37 %.

One of the trends in the application of DL methods in demand forecasting is the development of new hybrid DL models since it is possible to take advantage of the strengths of several models to form a new and more powerful model; we can cite [13], who developed a hybrid energy forecasting model named by CNN-GRU, based on sequential learning composed of convolutional neural network (CNN) and recurrent grid units (GRU). The performance of the proposed model (CNN-GRU) was compared with several state-of-the-art models namely linear regression, decision tree, support vector regression, CNN, LSTM and CNN-LSTM. The proposed hybrid model (CNN-GRU) had the lowest values of 0.22, 0.47 and 0.33 for the evaluation metrics MSE, RMSE and MAE, respectively, compared to the other models. In the same way [22], they combined the lightGBM and LSTM models to forecast the demand in the supply chain framework. The performance of the proposed model was compared with lightGBM model and LSTM model. The experiment showed that the combined model produces more accurate forecasts without wasting much time.

In general, in the area of electrical manufacturing, despite the cited advantages of DL methods, and in particular LSTM and GRU methods, their applications for demand forecasting in the manufacturing industry remain almost nil, raising questions about their abilities to handle highly nonlinear time series datasets in the real industrial world.

The objective of this paper is multiple, first is an experiment of real application of DL methods in the industrial field. Second, is the development of a new hybrid model that combines both LSTM and GRU models named deep LSTM-GRU more powerful. The proposed method can automatically configure the hyperparameters of the best-performing forecasting model on a time series data using the gridsearchCv technique. In order to demonstrate the performance of the new method, we compared it with the single and multi-layer deep learning LSTM models and the single and multi-layer GRU model using real data of an electrical product of a Moroccan company. The results of the experiment confirmed the robustness of the new structure in terms of accuracy and performance.

3. Methodology

3.1 Workflow

The objective of this study is to develop a hybrid model composed of the LSTM and GRU models that can compute an accurate forecast of the demand for electrical products. The proposed method is primarily based on the optimization of hyperparameters using the gridsearch technique. We also compare the performance of the new model to the simple LSTM, simple GRU, stacked LSTM, and stacked GRU methods. We exploit recent deep learning methods to identify the best-performing time series forecasting model. The methodology followed to build our new LSTM-GRU model is illustrated in Fig. 1, which describes the steps of construction of the method applied in this study.

3.2 Data description

3.2.1. Observational data

In general, the effective management of material and product inventories is a crucial aspect of any industry. Demand forecasting in the short term helps to ensure that supplies run efficiently and to adjust production capacities, such as the sizing of operational crews and machines. To achieve these goals, the company must predict future weekly demand based on previous sales. Currently, the company in question manually calculates the forecasts based on either the sales history or the marketing department since it is closer to the customer orders. In our application, we use the data of a Moroccan company named “Ingelec”, which specialized in electrical products manufacturing. We use the weekly sales history of the last five years of a product that makes the biggest turnover of the company.

Fig. 2 shows the product’s data from 2017 to 2021.

The time series of the quantity sold includes 224 weeks in total. Since the analyzed data covers a long period, the demand forecasting results produced using the new LD model based on this data can be powerful so that the company can plan its resources effectively.

3.2.2. Characteristics of Time series

a. Stationarity study

Time series are known as stationary or non-stationary. A time series is called stationary if its components did not change their characteristics over time. Among the most common components driving the non-stationarity of time series are seasonality and trend. The graphical representation of the linear values, moving averages and standard deviations of our time series in Fig. 3 shows that they are constant over time. In this case, we can conclude that our time series presents no trend or seasonality.

To prove this result we will apply a stationarity test on our time series using the augmented dickey-fuller (ADF) test [35]. The series is stationary if it

Received: March 15, 2022. Revised: April 12, 2022.

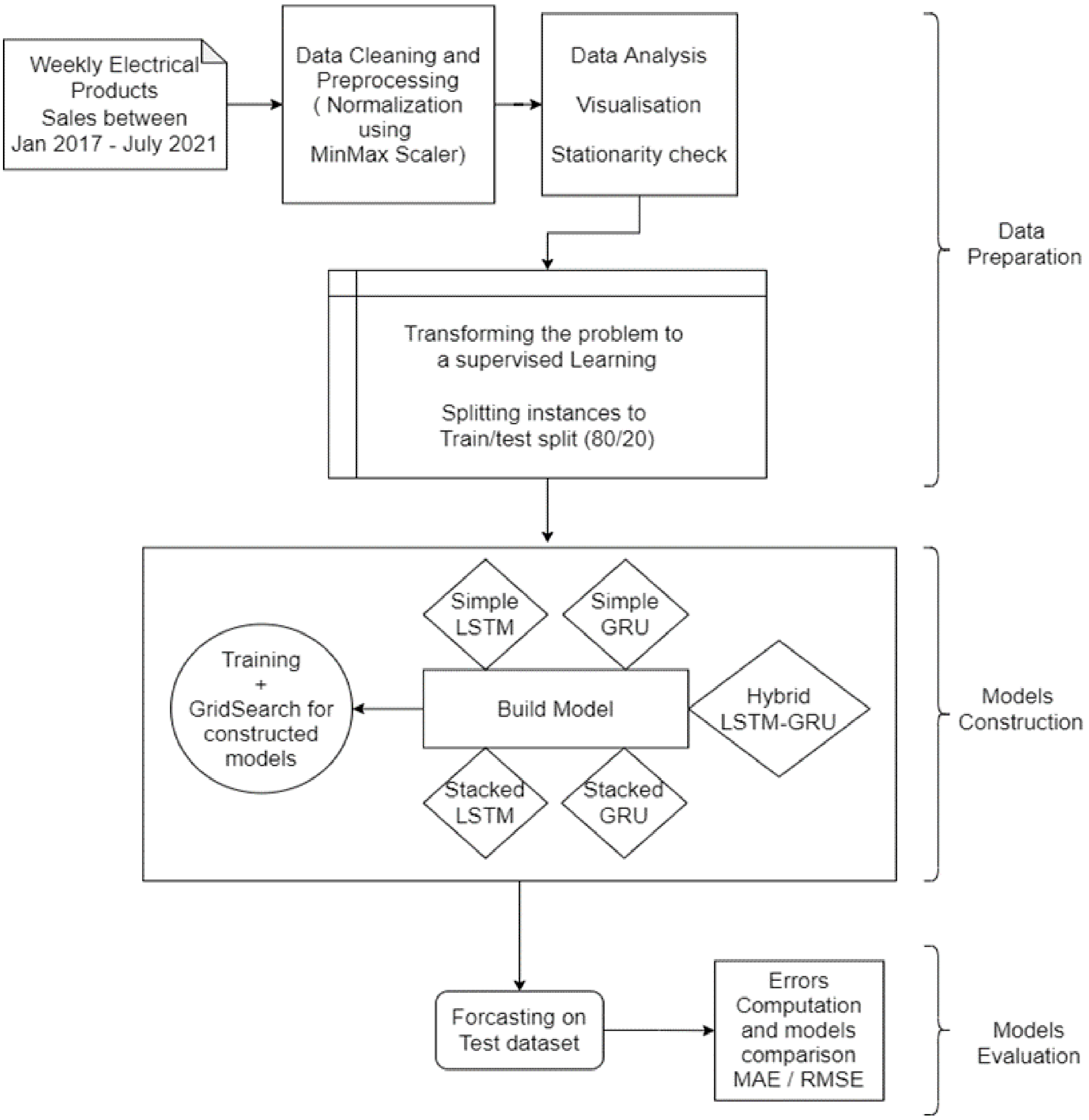

Figure. 1 The proposed methodology for demand forecasting using the time series forecasting approach

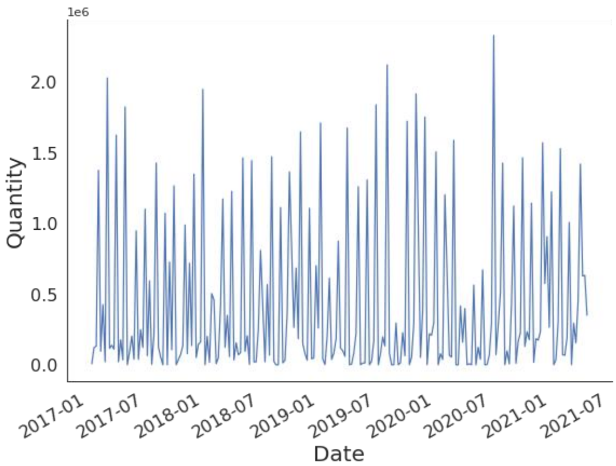

Figure. 2 Graphic presentation of the time series

doesn't have a unit root and the null hypothesis (H0) is rejected. As we can notice in Table 1, the value of p is relatively low and lower than the value

Table 1. ADF test results

	Value
ADF	-4.2112
P-Value	0.0006
Critical values 1%	-3.4617
Critical values 5%	-2.8753
Critical values 10%	-2.5741

alpha=0.05, therefore, the null hypothesis (H0) is rejected, that is to say, the series does not contain a unit root, we can deduce in this case that the series is weakly stationary.

b. Data description

Understanding the characteristics of our time series data is an essential step in a machine learning project. For this purpose, we used the boxplot technique, as this is an efficient way to deal with large data that are too difficult to manage, the main objective of describing the distribution of numerical

Received: March 15, 2022. Revised: April 12, 2022.

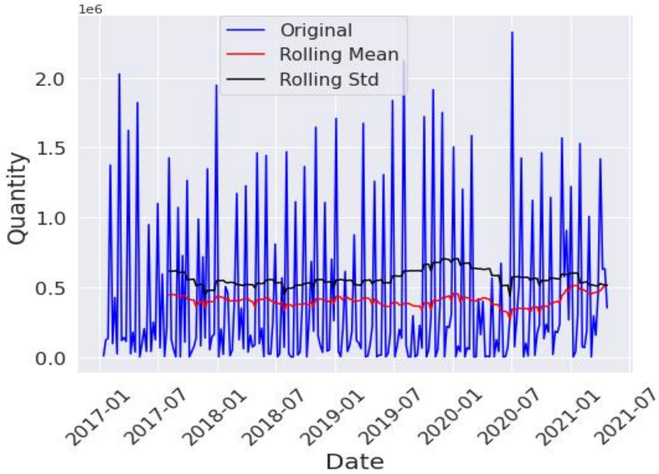

Figure.3 Trend plot of the time series

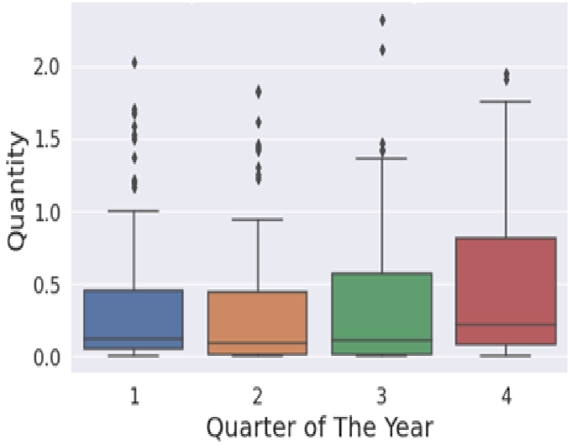

Figure. 4 Box plot of quarterly sales

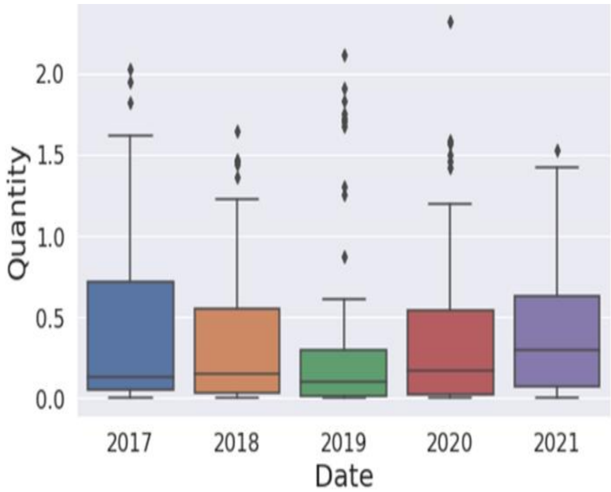

Figure. 5 Box plot of yearly sales

data is to visually extract some hidden statistical features by displaying the quartiles and means of the data.

Figs. 4 and 5 show that our time series data contain values that fall above the maximum interquartile range values, called outliers. An outlier is an observation that contrasts greatly with other measured observations of the same phenomenon. The presence of these outliers in our time series is due to the variability of demand in the market for this type of product. For this reason, we will keep them in our study even if they make the forecasting operation more complex.

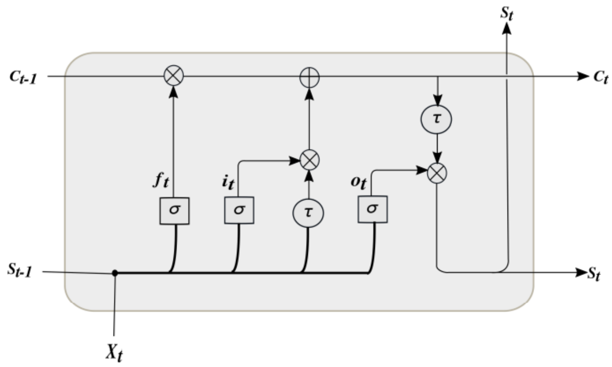

Figure. 6 LSTM block, where f_t, i_t, o_t are forget, input, and output gates respectively

3.3 Reference models

3.3.1. LSTM

The LSTM model is a variant of the RNN, developed by Horchreiter and al [36] in 1997, to solve the leakage gradient problem in the case where long-term context-dependency storage is required [37]. The key structure of the LSTM is a memory cell designed throughout the chain to consider the entire data sequences, and each time, depending on the need, it can either add or remove the incoming information by opening or closing the data stream as shown in Fig. 6.

The LSTM structure is regulated by three gates which are the means for determining the information to keep, eliminate and output each cell, these are the forget, update and output gates.

If (X_t) is an input at time *t*, and (S_{t-1}) is the hidden state of the previous time introduced in the LSTM block, the state (S_t) at time t will be calculated in the following way:

- The first step starts with the forgetting gate which decides which information should be discarded from the cell according to the following equation:

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

- The second step is to determine the information to be stored in the cell state through two steps: First, the input gate layer (i_t) updates the incoming values, according to the following equation:

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

Secondly, the *tanh* layer creates a vector of values named C, described as follows:

$$\tilde{C}_t = tanh\{W_C.(x_t, h_{t-1}) + b_C\} \quad (3)$$

- Then, the update of the previous cell state C_{t-1} must be established, to produce the new cell state C_t according to the following equation:

$$C_t = f_t \otimes C_{t-1} + i_t \otimes \widetilde{C_t} \quad (4)$$

- And then at the end, the output gate (σ_t) decides which piece of the cell state is going to be produced for the output, then for the output values to be filtered between -1 and 1, the cell state must be subjected to a *tanh* layer as follows:

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

$$h_t = o_t \otimes \tanh(C_{t-1}) \quad (6)$$

3.3.2. GRU

GRU is another variant of RNN, which was proposed by Cho et al [38]. GRU is based on the LSTM method but its internal structure is simpler due to the absence of a memory cell [38], which allows GRU to have faster learning than LSTM with less computational power. Contrary to the LSTM, the GRU contains two gates. The reset gate R defines how to combine the new input with the information from the previous memory, and then decide which ones to forget. This gate is calculated according to the following equation:

$$R = sig(W_R c_{t-1} + V_R x_t) \quad (7)$$

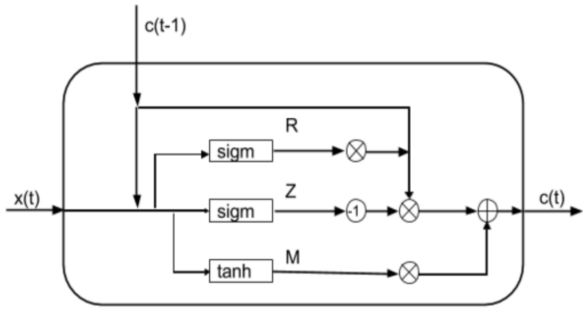

Figure. 7 Structure of GRU cell

Table 2. Data description

	Quantity
count	220.0000
mean	418353.8409
std	562274.9240
min	0.0000
25%	37661.0000
50%	148115.5000
75%	578090.2500
max	2325298.0000
Missing Values	0

The update gate Z defines the amount of previous memory information to keep, update and then move to the next state [39], as shown in Fig. 7.

This door is calculated as follows:

$$Z = sig(W_z c_{t-1} + V_Z x_t) \quad (8)$$

Then, based on the new input and the cache state of the past, a new element M is calculated according to the equation:

$$M = \tanh(W_M(C_{t-1} * R) + V_M x_t) \quad (9)$$

At the end, the hidden state is generated from the past hidden input and the new memory generated according to the following equation:

$$C_t = (C_{t-1}) * (1 - Z) + Z * M \quad (10)$$

Where *W* are the weight matrices, *V* are the parameter vectors, x_t is the input, C_t is the current output and C_{t-1} is the previous output.

4. Experiment

4.1 Data preprocessing

4.1.1. Data normalisation

The data used for this research contains no missing values. We did not use any noise reduction

Received: March 15, 2022. Revised: April 12, 2022.

or smoothing techniques on the data to keep the characteristics of the real industrial world data. In addition, we kept the existing outliers in our series as important data which increases the prediction complexity for the model to be built. See Table 2.

To facilitate the learning of the network, the formatting of the data values in a common small value scale is crucial in an ML project. For this reason, we used the MinMaxScaler method of Scikit-learn to perform this operation [40].

Let a time series of length N be represented as $\{(S(t_i), i= 1, 2..., N\}$,

The equation of the Min-Max Normalization is the following:

$$s(t_i) = \frac{s(t_i)-\min(s)}{\max(s)-\min(s)} \tag{11}$$

Where $s(t_i)$ represents the normalized value, S the observed values in the set and Min and max are the minimum and maximum values of x.

4.1.2. Transforming the problem into a supervised learning problem

For a supervised learning project, the dataset must have two essential elements, namely the features x and the target variable(s) y. Therefore, we need to reshape the time series data into a set of input instances and an output. The generation of the instances is done according to the previously defined offset size. This task is necessary to use deep learning methods. We have tested several lag values and have found 9 as the lag value giving the best performance of the proposed models.

4.1.3. Splitting instances into train and test sets

In this step, we divide the data into two independent parts, the training set presents 80 % and the test set 20 %. The training set is considered as the part on which the model learns. The test set is then used to evaluate the performance of the prediction results produced by the built model.

4.2 LSTM-GRU model framework and parameters selection

4.2.1. Model training parameters regularisation

In the field of machine learning, hyperparameter tuning is the most crucial step in the development of an ML model. A hyperparameter is a value that controls the learning process of a given algorithm. Hyperparameters differ from one algorithm to another. Hyperparameter optimization consists of using an optimization function (hyperparameter optimization) that can search for the combination of hyperparameter values that allow the model to achieve the highest score [41]. In this work, we use the gridsearch technique to configure the new proposed model (LSTM-GRU) that we will apply later to predict the demand. This method allows to automatically define the best parameterization of the model, based on a series of values of each hyperparameter defined before.

Table 3. Values specified for each hyperparameter

Hyperparameter	Values
Epochs	[50, 100, 200, 500]
Dropout_rate	[0.3, 0.5]
Batch_size	[5, 10, 30, 50]
First_layer	[32, 64, 128]
Second_layer	[32, 64, 128]
Learning rate	[0.01, 0.001, 0.0001]
Activation Function	['tanh', 'relu']

In this paper, we will try to optimize the following hyperparameters for the LSTM-GRU model:

- Epoch size: presents the number of learning iterations to be performed on the data set [42]. It is crucial to choose the epoch size carefully to have a performing model, because if it is too small, the model may suffer from the problem of under learning (i.e., underfitting), otherwise, if the number of epoch sizes is too large, the model will be overlearned.
- Batch size: represents the number of training examples that are shown to the algorithm in an epoch before the calculation of weights and parameters of the model.
- Number of neurons: this is a very important hyperparameter to specify, but it is very difficult to specify the optimal number of neurons for each layer. It is important to know that if the number of neurons is very low the model will be unable to keep all the important information necessary to make an accurate forecast. On the other hand, if the number of neurons is very high, the model may adapt too well to the training data which may cause a false generalization of the predictions.
- Optimizer: is an algorithm that allows you to link the loss function to the parameters during the model training process by modifying the weights, to indicate at the end the weights that allow your model to be as accurate as possible.
- Activation function: is a mathematical function that is applied to a signal. Its role is to allow or forbid the passage of information according to a predefined stimulation threshold.

 DOI: 10.22266/ijies2022.0630.51

- Learning rate: is a very important hyperparameter to obtain a high-performance model. It allows the adjustment of the magnitude of the modification of the model weights.
- Dropout rate: The application of the dropout-rate favors the independent extraction of the most general features. It consists in randomly deactivating the neurons of the same layer, i.e. setting the output value of the activation function to 0.

The list of search hyperparameter values is shown in Table 3.

4.2.2. Training model

According to the literature, there is no superior model in all cases of time series forecasting. The performance depends on the nature of the data, the domain of application, and the desired horizon for the forecast. For this reason, to justify the superiority of the new LSTM-GRU model parameterized with the gridsearch method, we compare the forecasts produced by the new model with those of four deep learning models namely simple LSTM, stacked LSTM, simple GRU, stacked GRU configured with the same gridsearch method. Following the training of the models, the best model was the hybrid LSTM-GRU model whom values of the hyperparameters specified by the gridsearch method are presented in Table 4. While those of the other comparison models are described in Table 5.

4.2.3. Model evaluation metrics

The observed and predicted data are continuous values, for this reason, we choose the mean absolute error (MAE) and the root mean square error (RMSE) to measure the differences between the predicted and observed demands.

(MAE) is defined according to the following equation:

$$MAE = \frac{1}{n} \sum_{t=1}^{n} \frac{|\widehat{y_t} - y_t|}{\frac{|\widehat{y_t}| + |y_t|}{2}} \tag{12}$$

Table 4. The hyperparameters for the best resulted LSTM-GRU model

Hyperparameters	**Values**
Batch size	2
Epochs	100
Dropout rate	0.3
Number_neurons	64
Activation_fonction	relu
Learning_rate	0.01
Second_neurones	32

Table 5. The hyperparameters values for the obtained resulted

Algorithm	**The best parameters**
Simple LSTM	batch_size= 5 Epochs= 150 Dropout_rate= 0.3 Neurones= 64 Learning_rate= 0.01 Activation= tanh Optimizer = adam
Stacked LSTM	batch_size= 5 Epochs= 200 dropout_rate= 0.3 Neurones= 64 Second_neurones= 64 lr= 0.01 Activation= relu
Simple GRU	Batch_size= 5 Epochs= 100 Dropout_rate= 0.3 Neurones= 128 lr= 0.01 Activation= relu
Stacked GRU	batch_size= 5 Epochs= 100 Dropout_rate= 0.3 Neurones=64 Second_neurones= 64 lr=0.01 Activation= relu

(RMSE) is represented by:

$$RMSE = \sqrt{\frac{1}{n} \sum_{t=1}^{n} (y_t - \widehat{y_t})^2} \tag{13}$$

5. Results and discussion

In this section, we discuss the performance comparison between the new model (LSTM-GRU) proposed in this study and the other models developed in this study, namely simple LSTM, simple GRU, stacked LSTM, and stacked GRU. Knowing that we used the gridsearch method to search for the most optimal hyperparameters values automatically with which the applied models can perform better. The optimized values for the new proposed model and the comparison models are presented in the previous section in Table 4 and 5 respectively.

This section will be organized as follows: At first, we present the performance measurement results of the developed models on the training and test sets of our time series. However, our analysis of the results will focus on just the test set. In particular, we compare the accuracy of the proposed LSTM-GRU model against the comparison models by presenting

Received: March 15, 2022. Revised: April 12, 2022.

the actual and projected weekly data produced by each model. Finally, we show that the results obtained in this study are consistent with related work in this research.

5.1 Evaluation models

The measurements of the errors made by each model during the forecast calculations are presented in Table 6. Comparing the error values measured by RMSE and MAE, it is clear that, the proposed new model LSTM-GRU is the best performing as it produces the lowest error values in the test, RMSE= 366633.28 and MAE= 237682.56, followed by the stacked GRU model with RMSE= 378104.7181 and MAE= 244985.8637. The simple GRU and stacked LSTM models produced forecasts with the same performance, due to the errors that are very close. In the end, we find that the least performing model is simple LSTM by committing the largest number of errors.

Table 6. MAE and RMSE for all methods

Method	Train		test	
	MAE	RMSE	MAE	RMSE
Simple LSTM	93908.39	164512 .97	273472 .03	410163 .28
Simple GRU	143986.0 1	229463 .22	263158 .04	389118 .28
Stacked LSTM	154067.8 7	250294 .49	262748 .03	391668 .18
Stacked GRU	162577.5 5	266106 .52	244985 .86	378104 .72
Our Method	198977.7 9	295721 .83	237682 .56	366633 .28

5.2 Comparative analysis

To further demonstrate the computational efficiency of forecasts by our proposed hybrid LSTM-GRU model, we plot the actual and projected demand forecasts by our new model and stacked GRU which is the second-best performing technique among all other methods in terms of RMSE and MAE. See Fig. 8.

We can notice that the newly developed LSTM-GRU and the stacked GRU model have the ability to predict peaks and zero values of demand, but the LSTM-GRU model is the most accurate since it can track more the variation of demand.

We can conclude in the framework of our study, that our proposed model LSTM-GRU which combines the two models of DL, LSTM and GRU is more powerful than the model LSTM and GRU.

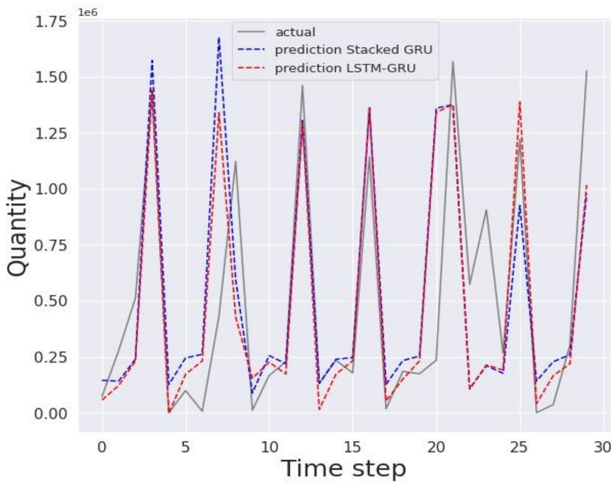

Figure. 8 Actual data vs prediction using stacked GRU and LSTM-GRU

This result is consistent with other recent works that confirm the power of hybrid models over simple DL models for time series forecasting in the industrial domain, such as [13] which developed the hybrid CNN-GRU model based on convolutional neural network (CNN) and recurrent grid units (GRU) for short-term residential energy forecasting. The authors confirmed that the hybrid model has the ability to produce more accurate forecasts compared to the CNN model and GRU model. Similarly, [22] who combined the lightGBM and LSTM models to forecast demand in the supply chain framework. And they confirmed that the forecasting performance produced by the hybrid model far exceeds the lightGBM and LSTM models.

6. Contribution

This work presents the first experience of DL methods in the context of the industrial supply chain of a Moroccan company specialized in electrical products manufacturing. The objective was to improve the weekly forecasting accuracy of a product that makes the big turnover for the given company by developing a hybrid DL model called LSTM-GRU configured automatically using the gridsearch method. The data used in this research are the real sales history of this product characterized by the presence of zero values and outliers. We didn't apply any smoothing or noise reduction method to keep the characteristics of the real industrial world data. In addition, we kept all the outliers and null values existing in our series, as important data which increases the prediction complexity for the model to be built. The new LSTM-GRU model has been shown to produce more accurate forecasts than the DL models: Simple LSTM, Simple GRU, stacked LSTM, and stacked GRU.

 DOI: 10.22266/ijies2022.0630.51

7. Conclusion

The objective of this work was to build a hybrid LSTM-GRU model capable of producing the best possible weekly forecasts for a Moroccan industrial company, using real data from the sales history of a product with the highest turnover. We used the grid search method to select the best hyperparameter combinations of the model to more appropriately capture the characteristics of the time series.

To demonstrate the superiority of the constructed model, we applied the gridsearch method on the models used for comparison in this study, namely simple LSTM, simple GRU, stacked LSTM, and stacked GRU. We used the MAE and RMSE error calculation indices to evaluate the performance of each model. The results show that the newly developed model has the lowest error measures: MAE = 237682.56 and RMSE= 366633.28, compared to the other applied models. This proves that the hybrid model composed of LSTM and GRU can produce more accurate and efficient predictions than the simple model. In our future work, we can apply the constructed model for demand forecasting in another SCM domain. We can also improve the accuracy of the proposed model by further improving the model's hyperparameters.

Conflicts of interest

The authors declare no conflict of interest.

Author contributions

Conceptualization, Aicha El filali and Aissam Jadli; methodology, Aicha EL FILALI; software, Aissam jadli and Aicha EL Filali; validation, El Habib Ben lahmer, Sanaa El filali, and Aicha El filali; formal analysis, Aicha EL filali; data source, Ingelec enterprise; writing—original draft preparation, Aicha El filali; writing—review and editing, Aissam Jadli; visualization, Aissam Jadli; supervision, El Habib Ben lahmer, Sanaa El filali.

Acknowledgments

The authors would like to thank Mr. Saidou Ahmed and Mr. Bereji Hassan for their linguistic assistance in writing this document.

References

[1] H. Haberleitner, H. Meyr, and A. Taudes, "Implementation of a demand planning system using advance order information", *International Journal of Production Economics*, Vol. 128, No. 2, pp. 518–526, 2010, doi: 10.1016/j.ijpe.2010.07.003.

[2] P. Chandra and M. L. Fisher, "Coordination of production and distribution planning", *European Journal of Operational Research*, Vol. 72, No. 3, pp. 503–517, 1994, doi: 10.1016/0377-2217(94)90419-7.

[3] W. E. Sasser, "Match Supply and Demand in Service Industries", *Harvard Business Review*, Nov. 01, 1976. Accessed: Jul. 03, 2021. [Online]. Available: https://hbr.org/1976/11/match-supply-and-demand-in-service-industries.

[4] M. White, "Digital workplaces Vision and reality", *Business Information Review*, Vol. 29, pp. 205–214, 2012, doi: 10.1177/0266382112470412.

[5] S. Jeble, S. Kumari, and Y. Patil, "Role of Big Data in Decision Making", *Operations and Supply Chain Management: An International Journal*, Vol. 11, No. 1, pp. 36–44, 2017, doi: 10.31387/oscm0300198.

[6] I. Affia, L. P. E. Yani, and A. Aamer, "Factors affecting IoT adoption in food supply chain management", In: *Proc. of 9th International Conference on Operations and Supply Chain Management*, 2019.

[7] M. Christopher and M. Holweg, "Supply chain 2.0 revisited: a framework for managing volatility-induced risk in the supply chain", *International Journal of Physical Distribution & Logistics Management*, Vol. 47, No. 1, pp. 2–17, 2017, doi: 10.1108/IJPDLM-09-2016-0245.

[8] J. S. Armstrong, "The Ombudsman: Research on Forecasting: A Quarter-Century Review, 1960–1984", *Interfaces*, Vol. 16, No. 1, pp. 89–109, 1986, doi: 10.1287/inte.16.1.89.

[9] D. S. Levi and Y. Zhao, "Safety Stock Positioning in Supply Chains with Stochastic Lead Times", *Manufacturing & Service Operations Management*, Vol. 7, pp. 295–318, 2005, doi: 10.1287/msom.1050.0087.

[10] E. Chong, C. Han, and F. C. Park, "Deep learning networks for stock market analysis and prediction: Methodology, data representations, and case studies", *Expert Systems with Applications*, Vol. 83, pp. 187–205, 2017, doi: 10.1016/j.eswa.2017.04.030.

[11] H. Abbasimehr, M. Shabani, and M. Yousefi, "An optimized model using LSTM network for demand forecasting", *Computers and Industrial Engineering*, Vol. 143, 2020, doi: 10.1016/j.cie.2020.106435.

[12] M. A. Villegas, D. J. Pedregal, and J. R. Trapero, "A support vector machine for model selection in demand forecasting applications", *Computers & Industrial Engineering*, Vol. 121, pp. 1–7, 2018, doi: 10.1016/j.cie.2018.04.042.

[13] M. Sajjad et al., “A Novel CNN-GRU-Based Hybrid Approach for Short-Term Residential Load Forecasting”, *IEEE Access*, Vol. 8, pp. 143759–143768, 2020, doi: 10.1109/ACCESS.2020.3009537.

[14] A. R. S. Parmezan, V. M. A. Souza, and G. E. A. P. A. Batista, “Evaluation of statistical and machine learning models for time series prediction: Identifying the state-of-the-art and the best conditions for the use of each model”, *Information Sciences*, Vol. 484, pp. 302–337, 2019, doi: 10.1016/j.ins.2019.01.076.

[15] P. Zhang, E. Patuwo, and M. Hu, “Forecasting With Artificial Neural Networks: The State of the Art”, *International Journal of Forecasting*, Vol. 14, pp. 35–62, 1998, doi: 10.1016/S0169-2070(97)00044-7.

[16] K. K. Chandriah and R. V. Naraganahalli, “RNN / LSTM with modified Adam optimizer in deep learning approach for automobile spare parts demand forecasting”, *Multimed. Tools Appl.*, Vol. 80, No. 17, pp. 26145–26159, 2021, doi: 10.1007/s11042-021-10913-0.

[17] M. R. A. Naseri and B. R. Tabar, “Neural network approach to lumpy demand forecasting for spare parts in process industries”, In: *Proc. of 2008 International Conference on Computer and Communication Engineering*, pp. 1378–1382, 2008, doi: 10.1109/ICCCE.2008.4580831.

[18] K. N. Amirkolaii, A. Baboli, M. K. Shahzad, and R. Tonadre, “Demand Forecasting for Irregular Demands in Business Aircraft Spare Parts Supply Chains by using Artificial Intelligence (AI)”, *IFAC-PapersOnLine*, Vol. 50, No. 1, pp. 15221–15226, 2017, doi: 10.1016/j.ifacol.2017.08.2371.

[19] A. Sagheer and M. Kotb, “Time series forecasting of petroleum production using deep LSTM recurrent networks”, *Neurocomputing*, Vol. 323, pp. 203–213, 2019, doi: 10.1016/j.neucom.2018.09.082.

[20] S. Alouini and S. Colcagno, “Les réseaux de neurones récurrents : des RNN simples aux LSTM | OCTO Talks !”, Oct. 21, 2019. https://blog.octo.com/les-reseaux-de-neurones-recurrents-des-rnn-simples-aux-lstm/ (accessed Sep. 22, 2021).

[21] M. L. Shen, C. F. Lee, H. H. Liu, P. Y. Chang, and C. H. Yang, “Effective multinational trade forecasting using LSTM recurrent neural network”, *Expert Systems with Applications*, Vol. 182, 2021, doi: 10.1016/j.eswa.204.115199.

[22] T. Weng, W. Liu, and J. Xiao, “Supply chain sales forecasting based on lightGBM and LSTM combination model”, *Industrial Management and Data Systems*, Vol. 120, No. 2, pp. 265–279, 2019, doi: 10.1108/IMDS-03-2019-0170.

[23] S. S. Namini, N. Tavakoli, and A. S. Namin, “A Comparative Analysis of Forecasting Financial Time Series Using ARIMA, LSTM, and BiLSTM”, *arXiv:1911.09512 [cs, stat]*, Nov. 2019, Accessed: Sep. 23, 2021. [Online]. Available: http://arxiv.org/abs/1911.09512.

[24] N. U. Moroff, E. Kurt, and J. Kamphues, “Machine Learning and Statistics: A Study for assessing innovative Demand Forecasting Models”, *Procedia Computer Science*, Vol. 180, pp. 40–49, 2021, doi: 10.1016/j.procs.2021.01.127.

[25] A. M. N. C. Ribeiro, P. R. X. D. Carmo, I. R. Rodrigues, D. Sadok, T. Lynn, and P. T. Endo, “Short-term firm-level energy-consumption forecasting for energy-intensive manufacturing: A comparison of machine learning and deep learning models”, *Algorithms*, Vol. 13, No. 11, pp. 1–19, 2020, doi: 10.3390/a13110274.

[26] A. Kumar, R. Shankar, and N. R. Aljohani, “A big data driven framework for demand-driven forecasting with effects of marketing-mix variables”, *Industrial Marketing Management*, Vol. 90, pp. 493–507, 2020, doi: 10.1016/j.indmarman.2019.05.003.

[27] M. Khashei and M. Bijari, “A novel hybridization of artificial neural networks and ARIMA models for time series forecasting”, *Applied Soft Computing*, Vol. 11, No. 2, pp. 2664–2675, 2011, doi: 10.1016/j.asoc.2010.10.015.

[28] M. Khashei and M. Bijari, “A novel hybridization of artificial neural networks and ARIMA models for time series forecasting”, *Applied Soft Computing*, Vol. 11, No. 2, pp. 2664–2675, 2011, doi: 10.1016/j.asoc.2010.10.015.

[29] D. Ni, Z. Xiao, and M. K. Lim, “A systematic review of the research trends of machine learning in supply chain management”, *International Journal of Machine Learning and Cybernetics*, 2020, doi: 10.1007/s13042-019-01050-0.

[30] A. Aamer, L. E. Yani, and I. A. Priyatna, “Data Analytics in the Supply Chain Management: Review of Machine Learning Applications in Demand Forecasting”, *Operations and Supply Chain Management: An International Journal*, Vol. 14, No. 1, pp. 1–13, 2020.

[31] D. C. Park, M. A. E. Sharkawi, R. J. Marks, L. E. Atlas, and M. J. Damborg, “Electric load forecasting using an artificial neural network”, *IEEE Transactions on Power Systems*, Vol. 6,

Received: March 15, 2022. Revised: April 12, 2022.

No. 2, pp. 442–449, 1991, doi: 10.1109/59.76685.

[32] A. Sagheer and M. Kotb, "Time series forecasting of petroleum production using deep LSTM recurrent networks", *Neurocomputing*, Vol. 323, pp. 203–213, 2019, doi: 10.1016/j.neucom.2018.09.082.

[33] C. Heghedus, A. Chakravorty, and C. Rong, "Energy load forecasting using deep learning", In: *Proc. of 2018 IEEE International Conference on Energy Internet (ICEI)*, pp. 146–151, 2018, doi: 10.1109/ICEI.2018.00-23.

[34] J. Zheng, X. Chen, K. Yu, L. Gan, Y. Wang, and K. Wang, "Short-term Power Load Forecasting of Residential Community Based on GRU Neural Network", In: *Proc. of International Conference on Power System Technology (POWERCON)*, pp. 4862–4868, 2018, doi: 10.1109/POWERCON.2018.8601718.

[35] S. E. Said and D. A. Dickey, "Testing for Unit Roots in Autoregressive-Moving Average Models of Unknown Order", *Biometrika*, Vol. 71, No. 3, pp. 599–607, 1984, doi: 10.2307/2336570.

[36] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory", *Neural Computation*, Vol. 9, No. 8, pp. 1735–1780, 1997, doi: 10.1162/neco.1997.9.8.1735.

[37] R. Pascanu, T. Mikolov, and Y. Bengio, "On the difficulty of training recurrent neural networks", In: *Proc of the 30th International Conference on Machine Learning*, pp. 1310–1318, 2013, Accessed: Feb. 24, 2022. [Online]. Available: https://proceedings.mlr.press/v28/pascanu13.html.

[38] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling", *arXiv:1412.3555 [cs]*, 2014, Accessed: Feb. 24, 2022. [Online]. Available: http://arxiv.org/abs/1412.3555.

[39] G. M. U. Din and A. K. Marnerides, "Short term power load forecasting using Deep Neural Networks", In: *Proc. of 2017 International Conference on Computing, Networking and Communications (ICNC)*, pp. 594–598, 2017, doi: 10.1109/ICCNC.2017.7876196.

[40] J. Han, J. Pei, and M. Kamber, *Data Mining: Concepts and Techniques*, Elsevier, 2011.

[41] M. Claesen and B. D. Moor, "Hyperparameter Search in Machine Learning", *arXiv:1502.02127 [cs, stat]*, 2015, Accessed: Jan. 27, 2022. [Online]. Available: http://arxiv.org/abs/1502.02127.

[42] N. Reimers and I. Gurevych, "Optimal Hyperparameters for Deep LSTM-Networks for Sequence Labeling Tasks", *arXiv:1707.06799 [cs]*, 2017, Accessed: Jan. 27, 2022. [Online]. Available: http://arxiv.org/abs/1707.06799.