Machine Learning Applications in Supply Chain Management: A Deep Learning Model Using an Optimized LSTM Network for Demand Forecasting

Aicha El Filali[1*]   **El Habib Ben Lahmer[1]**   **Sanaa El Filali[1]**   **Mohammed Kasbouya[1]**
Mohammed Amine Ajouary[1]   **Saadia Akantous[1]**

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

Abstract: The covid 19 has caused a strong health and economic crisis, where the business environment is turning into a great uncertainty and customer demand has become more and more fluctuating. As a result, demand forecasting as well as the task of predictive analysis, holds a great attention in the supply chain, in order to meet the needs of customers, avoid any out of stock, and at the same time avoid a waste of resources. In this paper, we propose a deep learning method based on long-term memory multilayer networks (LSTM) for demand forecasting. Using the grid search method, our method has the ability to automatically select the most optimal forecasting model, considering different combinations of LSTM hyperparameters of a time series. The proposed model has shown strength in capturing the existing nonlinear features in time series data compared to some known time series forecasting methods derived from statistical and machine learning approaches, using historical sales data of a Moroccan pharmaceutical manufacturing company. These methods include exponential smoothing (ETS), autoregressive integrated moving average (ARIMA), recurrent neural network (RNN). The evaluation of the proposed method and the comparison methods was performed using the root mean square error (RMSE) and Symmetric mean absolute percentage error (SMAPE). The comparison of the test results showed that the proposed method is the best performing with RMSE of 4487.32 and SMAPE of 0.026 much better than those obtained by the other models.

Keywords: Demand forecasting, Time series forecasting, Statistical methods, Deep learning, Long short term memory, Exponential smoothing, Autoregressive integrated moving average, Recurrent neural network.

1. Introduction

Research in the field of supply chain management has shown the impact of demand on the entire business process. Demand is considered one of the most important pieces of information to be shared among supply chain actors [1]. Demand forecasting is the basis for all planning activities in the supply chain [2]. Accurate demand forecasting is a key factor in an efficient supply chain that improves customer satisfaction while avoiding stock-outs [3–5]. Traditional forecasting approaches have proven unable to accurately produce customer demand [9–12]. Most of the time real world data are time series

with non-linear characteristics. The most known statistical methods for time series, such as ARIMA (autoregressive integrated moving average), holt-winters moving average, exponential smoothing, are classical linear models that are appropriate and satisfactory in many applications, since they can model data and make predictions on relatively large time spaces (year, month...), provided that the time series is totally linear [13]. In order to overcome the limitations of classical linear models, non-linear models have been developed such as the ARCH (autoregressive conditional heteroscedastic), GARCH (general autoregressive conditional heteroscedastic) [14], However, these models cannot provide good forecasts, as they are unable to model

all the non-linear components existing in the time series.

Therefore, companies are looking for more advanced methods that allow them to be as close as possible to the customers' needs [15]. Machine learning techniques have proven to be powerful in the field of predictive analytics, particularly in the prediction of demand in the supply chain of various economic activities [12, 16–20].

Recently, applications of ML techniques in the supply chain domain for time series forecasting have been shown to be effective in various industries, such as furniture demand [12], energy demand [21], the demand for electricity [22], the demand for cash at ATMs [23], the demand for tourism [6], natural gas demand [24], etc.

Frequently used ML methods for time series forecasting are artificial neural networks(ANNs), support vector machines(SVMs), K-nearest neighbors (KNNs) and adaptive neuro-fuzzy inference system (ANFIS) [12].

Since, artificial neural networks (ANNs) are one of the nonparametric data-driven models [14], they are considered the most important of the nonlinear temporal models in the field of forecasting due to several properties, first they are less sensitive to model misspecification problems as most of the nonparametric methods, they are adaptive by nature, moreover they use only a linear number of parameters [25].

Recurrent neural networks (RNN) are a type of artificial neural network, they are designed to recognize sequential features and to predict the next most likely scenario. They differ from traditional ANNs in that they use feedback loops in the form of a cycle allowing connections between nodes of an RNN and signals to move in different directions [26]. In the literature, RNN is considered the most widely used and best performing ML technique for demand forecasting [27] , RNNs are used in many areas of the supply chain, such as [28, 29] have shown the superiority of RNNs in forecasting spare parts, as well as for [30], have shown the performance of RNNs in medium and long term predictions for electricity consumption, [24] compared RNNs to linear regression and extreme machine learning algorithms for forecasting natural gas demand as a function of other variables: past temperatures, time variables, which include markers for vacations and other casual events. The tests showed the superiority of RNN. However, research has shown that RNNs are not satisfactory in all cases in the supply chain [27], this is due to their short memory and the problem of the disappearance of the gradient, which leads to a difficulty of training [31].

To overcome these limitations, the most popular solution is to use the Long Short-Term Memory (LSTM) method [32], this type of RNN can detect the most relevant information from the data and then split the time series signal between what is important in the short term, and what is in the long term [31, 33], which makes it capable of exploiting forecasts over long periods of time [12, 34, 35], and that the LSTM is able to produce long-term forecasts because of its dependence on the past [36].

Although these applications give promising results of forecasts produced by LSTM, it is possible to produce better performing forecasts that are more suitable for the industrial environment.

In this paper, we build the most robust LSTM model possible using the gridsearch technique, which is a method of searching all configurations of hyperparameters to produce the most accurate results for a model. There are many publications of LSTM applications, but to our knowledge, none of them specifically address the use of this model in pharmaceutical demand forecasting. We will show the results of the implementation of the multilayer LSTM model combined with the gridsearch method to forecast demand based on the sales history of a pharmaceutical product, in the Moroccan industrial context. The main contributions of this article are summarized as follows:

1. Real data of sales of a pharmaceutical product since 2012 are considered as much as a time series. We first performed a statistical analysis to study the stationarity of our series.

2. The methodology for developing the proposed forecasting method based on multi-layer LSTM combined with Gridsearch method is presented in steps and validated with the sales data of a pharmaceutical product.

3. The effectiveness of the method has been compared with two statistical models namely ETS and ARIMA and with both single layer and multilayer RNN models and single layer LSTM model.

4. The results suggest that recent research expectations are consistent with the prediction results obtained using the proposed method.

The rest of this paper is organized as follows: in the second section, we present a review of related research, then in the third section we expose the methodology that we followed to build the proposed model. Then, the fourth section contains the steps followed to apply our proposed method on our dataset, as well as the methods used to perform the comparison of the results. Finally, we conclude with the analysis and comparison of the results between

the different methods and a suggestion for future research in the field of supply chain forecasting.

2. Related work

Machine learning has a significant impact on SCM operations [37], ML algorithms can provide more accurate demand forecasts, which can reduce order amplification or the bullwhip effect (BWE) and improve the efficiency of operations along the supply chain [11, 38, 39]. In addition, on the basis of our systemic review [40], The number of publications on ML applications in the field of demand forecasting has relatively increased in the last two years, which shows the importance given to the improvement of the supply chain through ML methods, and that neural network methods are the most used algorithms in this field, they represent 61% of the total applications studied.

These findings were supported by other systemic reviews of research in the ML discipline in the context of the supply chain. [41] are among the first researchers who have carried out the systemic research of ML works in the discipline of supply chain, they have shown by analyzing 123 articles published in five large databases during the period between (1998/01/01) and (2018/12/31) that 87% of the applications were through supervised learning and that among the 10 ML algorithms that were found to be frequently used the methods of neural networks and its variants occupy 54% of the global total of applications. In the same way, [42] established a systemic study of 79 articles on ML applications in the field of supply chain and more precisely demand forecasting, published during the last ten years and coming from five databases. The authors concluded that neural network-based applications represent 54% of the studies reviewed, 65% of which were applied in the industrial sector.

The fame of neural networks in the field of demand forecasting could be due to the excellent results of deep learning which has pushed more and more people to apply it in sales forecasting. The deep learning model is able to handle the complexity of non-linear data and automatically extract its features.

For this reason, it is easier to obtain good prediction results at the first training even with arbitrary parameters. Several works have confirmed this result, such as in the article [10], where the authors applied the LSTM model to predict the demand for 5 products in the context of Industry 4.0. The performance of the LSTM model was compared to ML models: random forest (RF) and extreme gradient boosting (XGBoost) and statistical models: SARIMAX and triple exponential smoothing (ETS).

All the models were applied with the standard parameter settings determined in their above mentioned libraries with some exceptions.

For the evaluation of the forecasting models, the authors considered RMSE (root mean square deviation), to calculate the errors of the models, and the performance score to measure the deviation of the forecasting errors from the best model of each product. The experimental measurements showed that the lowest RMSE values were obtained with the ETS, XGBoost, LSTM and MLP models, but the highest performance score was obtained by the LSTM model, 81.9%. Regarding the effort of implementing the models in practice, ML methods proved to be particularly inexpensive compared to statistical methods that require very complex data preparation. Therefore, the proposed LSTM model can be most recommended for sales forecasting in the Industry 4.0 environment.

Similarly, the authors applied the multi-layer LSTM model in the paper [43] for water demand prediction, using the water consumption history of the year 2017 and 2018 collected from 20 residential houses. The two models support vector regression and random forest were used for the comparison of the performance of the proposed model. Bayesian optimization and 10-fold cross validation were applied to find the minimum penalty of each model. To evaluate the performance of the models, the authors used RMSE, MAE and MAAPE. The measurements showed that the LSTM and SVR models had comparable MAE and RMSE values, however the MAAPE values reflect that the LSTM model is the best architecture with MAAPE= 0.69, compared to SVR which had MAAPE=0.92 and RF with a MAAPE=0.98 value.

Although the predictions produced in the first training by the deep learning method with arbitrary parameters are more accurate in most of the time, one can further improve this performance by optimizing the chosen parameters, although they are quite difficult to parameterize. As an example in [44], the researchers point out that optimizing the weights of each node can increase the performance of the DL model, on this basis they have developed a deep RNN model based on LSTM optimized by the ADAM approach, in order to increase the accuracy of the forecasting of the demand for automotive spare parts. Monthly sales data for the last four years were used in this application. The proposed model was compared to the SBA, Croston, SES and TSB technique. The experiment showed that the optimized technique proved to have high forecasting accuracy with minimum errors MSE=30.167 and ME= 0.455 compared to the other methods used. The application

of the proposed model on new car sales data, showed that it allows the reduction of both the stocking level and the backorders of spare parts.

The contributions of the present work lie in the development of a new multilayer LSTM model based on the Gridsearch method. The proposed method has the ability to automatically configure by traversing several combinations of LSTM hyperparameters that allow increasing the accuracy of the model predictions for a given time series. The applications of ML methods and specifically deep learning in the field of pharmaceutical industry are very rare, to our knowledge, this is the first application of the LSTM method for future demand forecasting in this field using real data of the sales history since 2012 of the most demanded product of a Moroccan company working in the pharmaceutical industry. The performance of the proposed technique was validated by comparing it with the statistical methods of forecasting time series namely ETS and ARIMA. And with the monolayer RNN, multilayer RNN and monolayer LSTM methods. We show through the performance measures of each model that our results are consistent with recent research using the LSTM method.

3.   Methods used

In this section we present the principles models used in this study.

3.1 ARIMA

The ARIMA model is a univariate model that seeks to describe a single variable as an autoregressive integrated moving average process.

The equation presents the mathematical expression of the autoregressive part.

$$x(t) = \sum_{i=1}^{p} \alpha_i x(t - i) \qquad (1)$$

Where:

t: The index represented by an integer,
$x(t)$: the estimated value,
p: the number of autoregressive terms
α: the polynomial related to the autoregressive operator of order p.
Equation (2) presents the estimate of the next value of the time series that depends on the errors of the previous forecasts.

$$x(t) = \sum_{i=1}^{q} \beta_i \varepsilon(t - i) \qquad (2)$$

Where:

q is the number of moving average terms

β is the moving average polynomial of order q
ε: the difference between the actual and estimated values of $x(t)$.

The ARIMA model (p and q) is the combination of Eqs. (1) and (2) presented as follows in Eq. (3):

$$x(t) = \sum_{i=1}^{p} \alpha_i x(t - i) \sum_{i=1}^{q} \beta \varepsilon(t - i) \qquad (3)$$

3.2 Exponential smoothing

Exponential smoothing methods that allow forecasts to be made based on the observation of a time series. These methods are often used in industry, being relatively basic and simple to implement, especially when the number of forecasts to be made is large.
Three types of exponential smoothing:

Simple exponential smoothing: which allows to locally adjust a constant to the time series.

- Double exponential smoothing: which adjusts a line.
- Holt-Winters exponential smoothing: which studies more complex functions (polynomial, periodic...).

The disadvantage of extended exponential smoothing is that the values chosen for ALPHA, BETA, and GAMMA greatly affect the accuracy, and finding the most accurate combination of three different smoothing constants, each of which can have a value between zero and one, may be a formidable task.

3.3 Artificial neural networks

Artificial neural networks are inspired by the design of the brain. It is a computational model that can be considered as formal models with equations and statements about the parts that are to be used. These models consist of a large number of basic computational neurons connected to each other in a complex communication network [53]. This is where a supervisor enters a teaching position and interacts with the learner, i.e., the network, and the environment. This is called supervised learning.

As shown in Fig. 1 above, the ANN is a direct feed system and consists of an input layer, a hidden layer with 3 cells with activation functions in each cell, and an output layer with an activation function that determines the output of the cell. An activation function works like a transformer in the network, where it takes input values and transforms them into output. In the figure above, we see a sigmoid function,
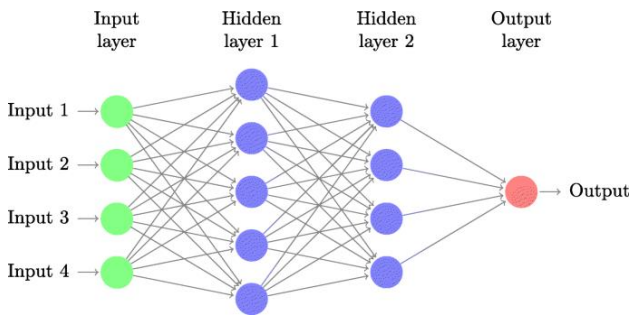
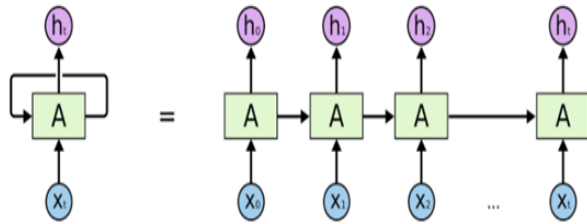Figure. 1 Artificial neural network architecture

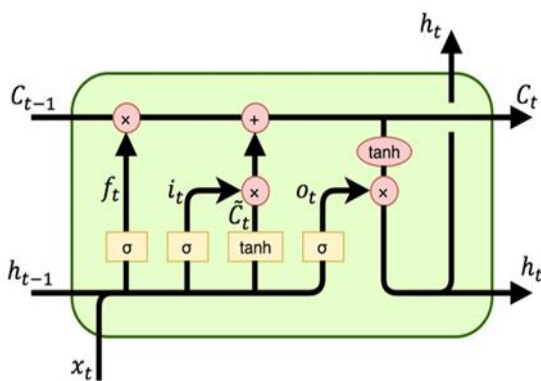
Figure. 2 Recurrent neural network architecture

Figure. 3 LSTM network's architecture

which is a special case of the logistic function used in logistic regression, to transform the outputs into binary outputs between 0 and 1. The hidden layer functions as a learning box and as a filter depending on the nature of the data and its structure.

3.4 Recurrent neural networks

RNN belong to the family of artificial neural networks with recurrent architecture.

Recurrence is the use of previous outputs as new inputs so that they are recurrent, i.e., they appear one or more times in the calculation of new outputs. A simple RNN takes as input not only the current input, but also previous information, through recurrent connections. This gives importance to previous events, which means that t-2 and t-1 will influence the decisions made at time t, see Fig. 2. The basic RNN is known as a vanilla RNN or SIMPLE RNN. RNN are sequential models that process one or more

sequential inputs of variables [54]. Their main meaning is that they can use all previous inputs for each output. so they are recurrent giving the notion of memory in the neural network [55].

3.5 Long short term memory

LSTM networks are a special form of RNNs that can handle both long and short term dependencies. They were first introduced in 1997 by Hochreiter and Schmidhuber [56], and subsequent work has improved and promoted them.

Traditional RNNs' structure theoretically allows for the control of long-term memory dependencies, but their impact in practice is limited [57]. As a result, RNNs' memory storage capacity is better suited to short-term sequences. The cell states and gate mechanism are added to the hidden layer based on traditional RNNs, so the gradient vanishing problem is largely addressed by its control gates. Furthermore, various control gates with distinct functions are employed to manage the past and present information whenever the historical message is transmitted to the neurons in the buried layer.

Fig. 4, shows the description of a LSTM memory block follows that of [58, 59].

The specific steps of the LSTM algorithm are as follows:

a. The memory cell reads the input as well as the prior hidden state, allowing long-term dynamic trends to be shown and redundant data to be removed. The following equation determines the forgetting gate:

$$f_t = \sigma\{W_f.(x_t, h_{t-1}) + b_f\} \qquad (4)$$

b. The first part of the model's input gate defines how much current data should be retained in the cell state:

$$i_t = \sigma\{W_i.(x_t, h_{t-1}) + b_i\} \qquad (5)$$

c. The second component entails constructing a new candidate vector to update the state, as shown in the equation below:

$$\tilde{C}_t = tanh\{W_C.(x_t, h_{t-1}) + b_C\} \qquad (6)$$

d. After that, the new state of the cell is constructed on the basis of the results of the last steps by designating the Hadamard product (by elements):

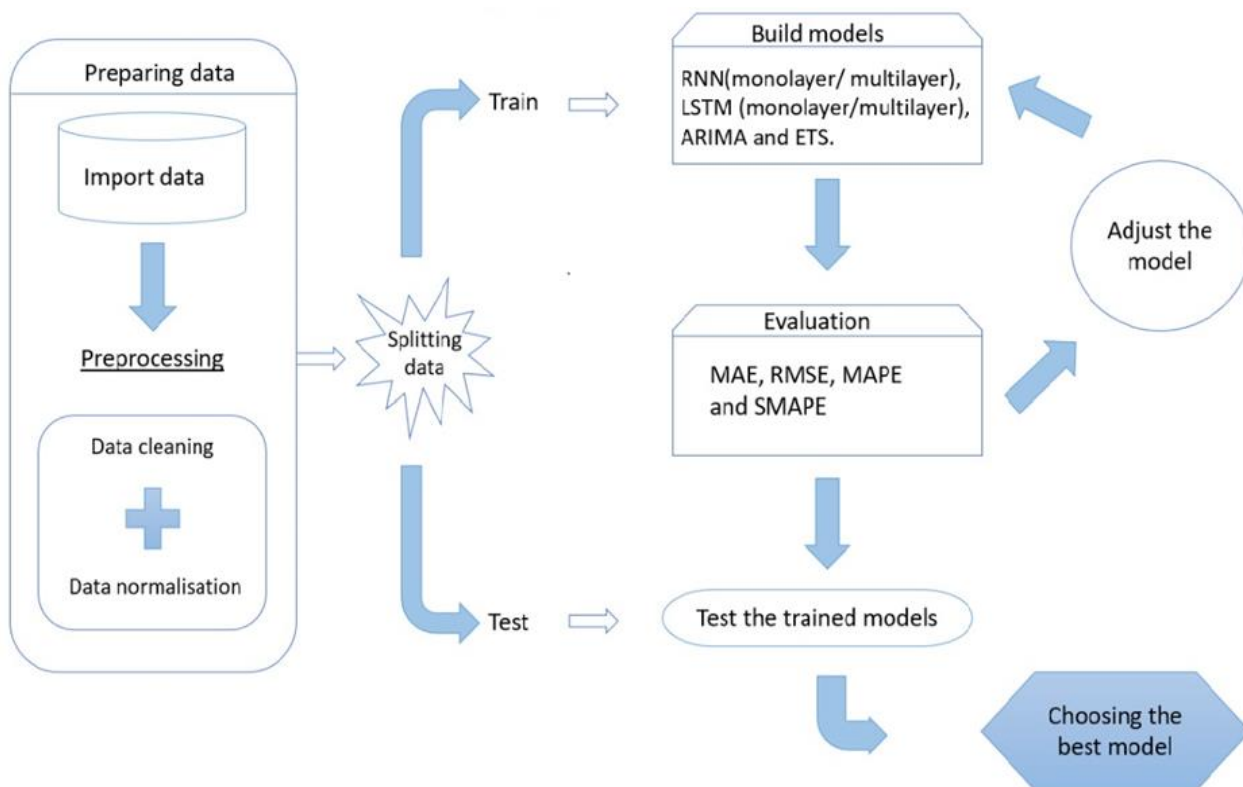$$\tilde{C}_t = tanh\{W_C.(x_t, h_{t-1}) + b_C\} \qquad (7)$$

Figure. 4 Flowchart of the proposed method

e. Finally, the output gate is updated and the final output is decided based on the updated state and the state of the output gate:

$$o_t = \sigma\{W_o.(x_t, h_{t-1}) + b_o\} \qquad (8)$$

$$h_t = o_t \otimes \tanh(C_{t-1}) \qquad (9)$$

In the above equations, the following notation is used:

a. x_t is the input vector at the current time step t.
b. W_C, W_f, W_i, W_o, are the weight matrices that are associated to the corresponding vectors. They can be decomposed into

c. $$\begin{cases} W_f = W_{fx} + W_{fh'} \\ W_i = W_{ix} + W_{ih'} \\ W_C = W_{Cx} + W_{Ch'} \\ W_o = W_{ox} + W_{oh'} \end{cases} \qquad (10)$$

d. b_c, b_f, b_i, b_o are indicators of bias.
e. f_t, o_t, i_t , and are vectors of door of forgetting, door of entry and door of exit.
f. C_t and $\tilde{C}_t$ are vectors for cell states and candidate values.
g. h_t is a vector for the output of the LSTM layer.
h. $\sigma(.)$ and $tanh(.)$ are the sigmoid function and the hyperbolic tangent function respectively.

4. Proposed methodology

Designing a machine learning model for time series forecasting can be a time-consuming procedure and require a structural construction approach. To choose the optimal time series forecasting model, we use modern deep learning algorithms. The proposed methodology for pharmaceutical time series demand forecasting is illustrated in the Fig.4 below which describes the steps of the proposed method.

4.1 Data preprocessing

The data we use for this study present the sales history of a pharmaceutical product from a Moroccan industrial company. The monthly time series we use starts from January 2012 to December 2020. Before embarking on the exploration part of a time series, it is important to edit the raw data and prepare it well. Cleaning and normalization are two main steps in pre-processing the data.

4.1.1. Data cleaning

If the time series contains outliers or missing values, they are replaced using one of the appropriate techniques. Outliers or accidental values, present observations that are extremely different from other

values of the same variable. These data are erroneously included, detecting these values saves money, time and ensures that our data is reliable without errors present.

Our dataset does not contain missing values, but there are outliers due to the error of data entry. These values are the cause of a non-performing and/or far from optimal predictive model. From a semantic point of view, these detected values are not logical, so we have chosen to remove them.

4.1.2. Data normalisation

The goal of normalization is to keep the values of the source dataset in a common scale to facilitate the learning of the network, the data should take small values, most often between 0 and 1. To do this, we used scikit-learn MinMaxScaler method to perform this operation.

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:

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

Where $s(t_i)$ represents the normalized value and s the observed values in the set.

Min and max are the minimum and maximum values of x.

4.2 Modeling

4.2.1. Data transformation for supervised learning

Time series data is first molded into a set of instances with predetermined input and output properties. The instances are then separated into training and test sets. Construction of learning, testing and validation sets.

4.2.2. Training, testing, and validation sets

When working with a large data set, it's advisable to split it into three parts: a training set, a test set, and a validation set.

4.2.3. Search for hyperparameters

Finding the most optimal models is a time-consuming process that leads to finding methods to find the best configuration where it achieves the best performance on the evaluation metrics chosen for the training and test sets. This makes sense, as it reduces the time spent on manual trial and error and instead

creates a computationally efficient mechanism for receiving the optimal configurations.

Hyperparameters are those parameters that are independent of the data and can be thought of as the buttons of a black box. They are used to control the model and are defined as the parameters that cannot be learned directly from the data [60].

Grid search is a method that helps us find the hyperparameters of interest. It consists of searching all possible configurations to produce the most accurate results. However, grid search can be computationally demanding and sometimes requires a lot of time and computing resources. The larger the number of parameters tested by the grid search, the more demanding it becomes. In this paper, we will use the following hyperparameters for LSTM:

- Batch size: number of training examples to pass in an epoch, this value will be set to 1.
- Time-steps: the number of steps in time taken by the model where it memorizes the number of steps.
- Optimizer: an algorithm to optimize the weights to reduce the mean square error.
- Couche dense: A dense layer is the most frequently used layer. It is essentially a layer where each neuron receives input from all neurons in the previous layer - thus "densely connected". Dense layers improve overall accuracy and 5-10 units or nodes per layer is a good base. Thus, the shape of the final dense layer output will be affected by the number of neurons/units specified.
- Dropout rate: single LSTM layer must be accompanied by an exclusion layer. By bypassing randomly selected neurons during training, this layer eliminates overfitting and reduces sensitivity to individual neuron weights.
- Epoch: this hyperparameter defines the number of full iterations of the dataset to be run. Although theoretically this number can be set to an integer value between one and infinity, it should be increased until the validation accuracy starts to decrease even if the learning accuracy increases (and thus risk overfitting).

4.3 Configuration of results

To decide which model is best, we need to be able to evaluate their performance. To evaluate the performance of the proposed approach, has been compared with two statistical models namely ETS and ARIMA and with both single and multi-layer RNN models and single layer LSTM model. For the
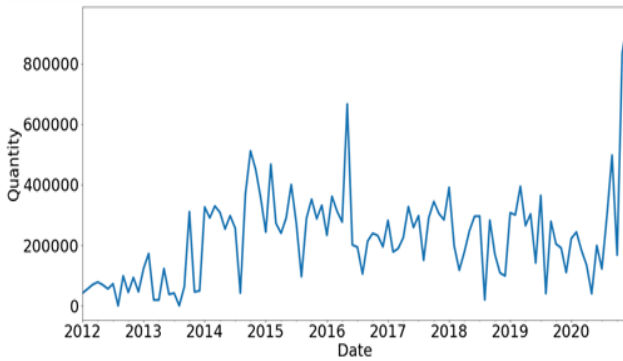
Figure. 5 Graphic presentation of the time series

Table 1. ADF test results

	Value
ADF	-9.1759
P-Value	$2.3207e^{-15}$
Critical values 1%	-3.4314
Critical values 5%	-2.8620
Critical values 10%	-2.5670

Table 2. List of hyperparameters used in grid search

Hyperparameter	Value
Units	64
Dropout_rate	[0.0, 0.1, 0.2, 0.3, 0.4, 0.5]
Activation	['relu', 'tanh', 'sigmoid', 'hard_sigmoid']
Batch_size	[8, 16, 32, 48]
Epochs	[50, 100, 150, 200, 250]
Optimizer	['Adagrad', 'Adadelta', 'Adam', 'Adamax', 'Nadam']

evaluation of all models, we use the performance measures RMSE and SMAPE [62]. SMAPE is defined by:

$$SMAPE = \frac{1}{n} \sum_{t=1}^{n} \frac{|\widehat{y_t} - y_t|}{\frac{|\widehat{y_t}| + |y_t|}{2}} \qquad (12)$$

The RMSE is expressed as follows:

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

Where y_t is the value taken by the explained variable at time t, $\widehat{y_t}$ is the value of the explained variable calculated by the model and n is the observations number.

5. Experiments

5.1 Data description

The data used is the monthly amount of product sales from 2012 to 2020 of a pharmaceutical company.

Every industry, in general, relies on the management of available materials and products. Determining future demand for raw materials and goods is the greatest strategy to manage the appropriate amount of these materials and items.

Companies must forecast future demand based on historical sales to attain this goal. The data utilized in this article is from a pharmaceutical company's sales. From January 2012 to December 2020, this data represents the monthly sales quantity for items. Predicting future demand with this information can assist the organization in improving its resource management strategies and efficiently planning raw material supply. Because the examined data spans a lengthy period of time, the resulting demand projection is likely to be accurate and cover the market's future demand behavior.

5.2 Data preprocessing

The most important phase in an ML project is the data preparation phase. In this step we have to do a thorough reading of our data, in order to understand the role, they play in the achievement of the prediction objective. The preparation of the data is a manual work, whose objective is to have at the end, a data set of good quality, appropriate to the modeling and to the realization of our objective. This step begins with the description of the data, the search for missing data, the elimination of useless data, the study of stationarity, the scaling of the data... etc.

After the analysis of the data, we did not detect any missing values, or outliers, which shows the reliability of the sources of the collected data.

5.2.1. Stationarity study

The notion of stationarity is an important factor in the modeling of time series. We can say that a series is stationary if the structure of the process in question remains the same over time, that is, if its characteristics: variance, expectation, variance, and autocorrelation do not vary over time. When this condition is not met, the series is considered non-stationary. Therefore, it is essential to be able to determine whether the series is stationary or not.

To measure the degree of stationarity of our time series, we used one of the unit root tests, the augmented dickey fuller (ADF) test method [63], for which the series is stationary if the null hypothesis (H0) is rejected when the process of the series does not have a unit root.

The values of the descriptive statistics of the series are presented in Table 2. As we can notice, the value of p is very low and lower than the value alpha

472

Table 3. The best resulted hyperparameters for LSTM model

Hyperparameter	Value
Lag	10
The number of hidden layers	2
Epochs	100
Batch size	2
Number of neurons per layer	64
Optimizer	Adam

= 0.05, therefore, we can reject the null hypothesis (H0) that the series presents a unit root, and retain that the series is low stationary. When the time series is stationary, the modelling process gets easier than in the opposite case, since we can predict the future based on past observations.

5.2.2. Data normalization

There are no missing values in the supplied time series. Furthermore, no noise reduction or data smoothing was applied to the series in order to preserve the properties of the real-world data. Because most forecasting techniques perform better with normalized data, we used the min-max normalization algorithm to normalize the data in this study.

5.2.3. Generation of train, test, and valid samples

Based on the selected offset size, the time series is transformed into a set of instances in input-output format. For an offset size of 10. Next, the created instances are divided into a training set and a test set.
a. Training data contains monthly sales volume covering the period from January 2012 to July 2017.
b. Test data, similar to the training set, covers the period from August 2017 to March 2019.
c. Validation data, covering the period April 2019 to December 2020

5.3 Generation of a list containing combinations of hyperparameters

Table 3, represents a list of determined values, for each hyperparameter, which will be used later in the configuration and training of each model.

5.4 Training model

From our literature review, we deduce that there is no one model that is superior to the others for time series forecasting, and that the performance of the model depends on the nature of the data, the context of application as well as the forecast horizon. For this reason, to show the efficiency of our proposed model, we apply other statistical and ML models, namely ARIMA, ETS, single and multilayer RNN and single layer LSTM. In order to obtain the best possible performance, it is important to note that we performed a search for the most optimal hyperparameters for each method used using the grid search method.

The hyperparameters of our constructed method are described in Table 3, while the other models are presented in Table 4.

An LSTM network is constructed and trained for each combination of hyper parameters. As an optimizer, the Adam algorithm [65] was utilized. In addition, for all developed LSTM networks, we used the mean square error (MSE) as the loss function. was used as an optimizer.

6. Results discussion

To prove the effectiveness of the proposed method for forecasting the demand of a pharmaceutical product, several comparison methods, including the statistical methods ARIMA and ETS and the methods RNN monolayer, RNN multilayer, LSTM monolayer and our proposed model LSTM multilayer. Knowing that we have applied the Gridsearch method for all the models used in this work in order to select automatically the adequate hyperparameters, which allows us to increase the efficiency of each model.

The results section will be organized as follows: First, we report the SMAPE and RMSE values calculated for each model. Then, we show the prediction performances through the presentation of the real and projected monthly data produced by each model. Finally, we compare the results on the prediction performance of each predictor ARIMA, ETS, single layer RNN, multilayer RNN, single layer LSTM to our proposed model LSTM multilayer.

6.1 Performance results

The comparison values of model performance measures for demand forecasting are presented in Table 4 respectively. By comparing the results of the RMSE and SMAPE error index values, we can conclude that the traditional forecasting approaches used for comparison, namely ARIMA and exponential smoothing have the largest error values compared to the ML methods. This shows that the time series data used in this case study have nonlinear characteristics that both ARIMA and ETS are unable to capture. It can also be seen that the proposed model

Table 4. The best parameters of the obtained models

Model	The best parameters
RNN	Lag: 10 The number of hidden layers: 1, 2 Epochs :50 Batch size:2 Number of units in the hidden layer (n): 64 Optimizer : Adam
LSTM	Lag: 10 The number of hidden layers: 1 Epochs :100 Batch size: 2 Number of units in the hidden layer (n): 64 Optimizer : Adam
ETS	Trend: Additive Seasonality: Additive Length of one seasonal period (s):20
ARIMA	Length of one seasonal period (s): 20 Order and seasonal Order: The best parameters obtained using the Auto-ARIMA function.

Table 5. Performance test results

Method	RMSE	SMAPE
ARIMA	13534	0.12
ETS	16653	-
RNN MONO	10925	0.068
RNN MULTI	5722.32	0.028
LSTM MONO	4707.78	0.027
LSTM MULTI	4487.32	0.026

LSTM multilayer has less error, by examining the values of RMSE and SMAPE compared to LSTM monolayer and RNN monolayer and multilayer. This shows that it is the best model to produce the best forecasting results.

6.2 Comparison forecasting

The results of the RMSE and SMAPE error evaluations for each model used in these studies, presented in Table 5, show that the ARIMA and ETS statistical methods have the highest error numbers compared to the ML methods. Several recent researches in the field of demand forecasting have confirmed the superiority of ML methods over statistical approaches such as [12, 48, 49, 52].

In this section, in order to show the computational efficiency of forecasting by our proposed method, we plot the actual and projected demand forecasts by the deep learning models developed using Gridsearch method, namely single-layer RNN, multilayer RNN, single-layer LSTM and multilayer LSTM.

- ▪ **RNNs monolayer**

RNN seems to be a more appropriate method for time series forecasting. However, it suffers from the problem of the disappearance and explosion of the gradient. Applied to our data, it obtained less accurate performances in terms of forecasting with a very high RMSE and MAPE error rate of 10925 and 0.068, respectively, compared to the single and multi-layer LSTM model.

- ▪ **RNN multilayer**

As shown in Table 5, the multilayer RNN model was able to produce predictions with 48% lower errors for RMSE and 59% lower errors for SMAPE than those produced by the single-layer RNN model. These results show that the deeper the layers, the better the model performs. However, we observe that the number of errors is still larger than for the multilayer LSTM model.

- ▪ **Monolayer LSTM:**

The values of RMSE and SMAPE for the single layer LSTM are relatively lower by 17% and 3% than that of multilayer RNN respectively, for this reason, the single layer LSTM method was able to produce predictions closer to the actual values compared to the multilayer RNN model, as shown in Fig. 9.

- ▪ **LSTM multilayer:**

The graphical presentation of the predictions produced by the multilayer LSTM model developed using the Gridsearch method fully overlaps with the real values. This result shows that this model performs better than the single layer LSTM, then the errors presented by the multilayer LSTM are less than 5% for the RMSE index and 4% for the SMAPE compared to the single layer LSTM.
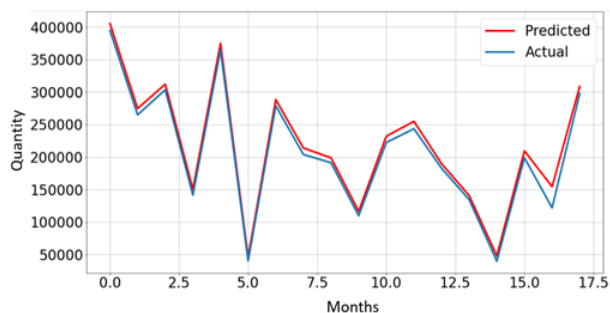

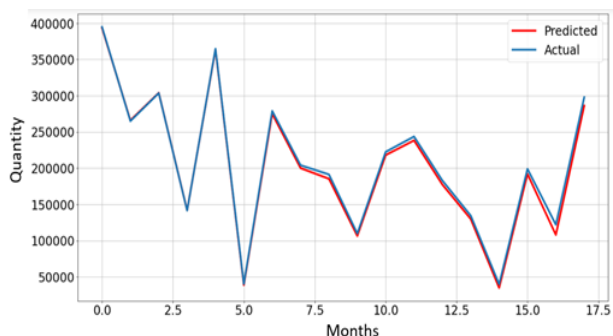Figure. 6 Monthly demand forecast using mono-layer RNN

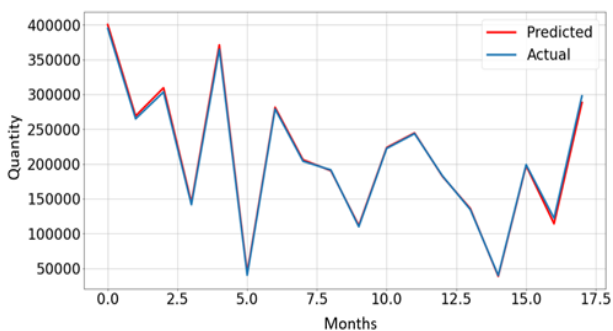Figure.7 monthly demand forecast using multi-layer RNN

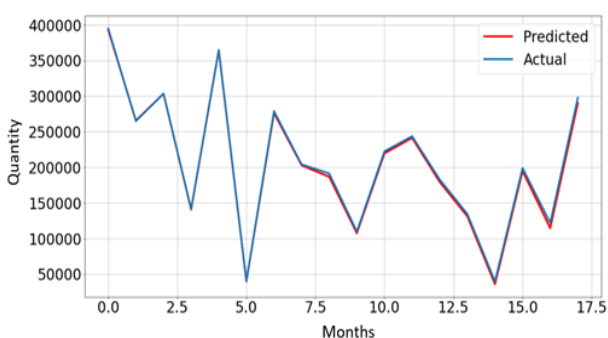Figure.8 Monthly demand forecast using LSTM monolayer

Figure.9 Monthly demand forecast using LSTM Multi-Layer

7. Contribution

This work presents the experimental application of ML methods for demand forecasting of the product with the highest turnover of a Moroccan company working in the pharmaceutical industry. Recent research on the application of ML techniques in the field of SCM has shown that the LSTM method produces good results in demand forecasting even with standard parameters predetermined in their above mentioned library [10, 43]. In practice, the increase in forecasting accuracy further improves the efficiency of SCM processes, therefore, the objective of this paper was to propose a new and improved LSTM model using the Gridsearch method that can automatically select the most suitable hyperparameters for the nature of the application's time series. The proposed method shows clear performance compared to statistical, single and multilayer RNN and single layer LSTM approaches.

8. Conclusion

The objective of this work was to build an LSTM model capable of producing the best possible forecasts, using real data from the sales history of a pharmaceutical product of a Moroccan company. We used the grid search method to select the best combinations of model hyperparameters and to better capture the characteristics of the time series.

In order to demonstrate the superiority of the constructed model, we applied the Gridsearch method on all the methods used for comparison in this study namely two statistical models: ARIMA and exponential smoothing and the ML models: the multilayer RNN model and the monolayer LSTM. We used the RMSE and SMAPE error calculation indices to evaluate the performance of each model. The results show that the proposed multilayer LSTM with Gridsearch method has the lowest error measures: RMSE= 4487.32 and SMAPE= 0.026, compared to the traditional approaches and the multilayer RNN and monolayer LSTM models. This proves that the fitted LSTM model is capable of producing more accurate and efficient predictions. In our future work, we can apply the constructed model for demand forecasting in another SCM domain, and we can also improve the accuracy of the proposed model by constructing a hybrid LSTM model with another ML or statistical method.

Conflicts of interest

The authors declare no conflict of interest.

Author contributions

Conceptualization, Aicha EL Filali, EL Habib Ben Lahmer and Sana El Filali; methodology, Aicha EL filali, El Habib Ben Lahmer, Sanaa El Filali, Mohammed Kasbouya, Mohammed Amine Ajouari and Saadia Akantous. software, Aicha EL Filali, Mohammed Kasbouya and, Mohammed Amine Ajouary; data source, validation, Aicha El Filali, EL Habib Ben Lahmer. writing— original draft preparation, Aicha El Filali, Mohammed Kasbouya, Mohammed Amine Ajouary and Saadia Akantous.

References

[1] M. Abolghasemi, E. Beh, G. Tarr, and R. Gerlach, "Demand forecasting in supply chain: The impact of demand volatility in the presence of promotion", *Computers and Industrial*

Engineering, Vol. 142, 2020, doi: 10.1016/j.cie.2020.106380.

[2] 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, Dec. 2010, doi: 10.1016/j.ijpe.2010.07.003.

[3] 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.

[4] B. Fleischmann, H. Meyr, and M. Wagner, "Advanced Planning", in *Supply Chain Management and Advanced Planning: Concepts, Models, Software and Case Studies*, H. Stadtler and C. Kilger, Eds. Berlin, Heidelberg: Springer, pp. 71–96, 2002. doi: 10.1007/978-3-662-10142-1_5.

[5] 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.

[6] J. Shahrabi, E. Hadavandi, and S. Asadi, "Developing a hybrid intelligent model for forecasting problems: Case study of tourism demand time series", *Knowledge-Based Systems*, Vol. 43, pp. 112–122, 2013, doi: 10.1016/j.knosys.2013.01.014.

[7] 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.

[8] "Introduction aux séries temporelles", *Le Data Scientist*, May 01, 2019. https://ledatascientist.com/introduction-series-temporelles/ (accessed Sep. 20, 2021).

[9] J. N. C. Gonçalves, P. Cortez, M. S. Carvalho, and N. M. Frazão, "A multivariate approach for multi-step demand forecasting in assembly industries: Empirical evidence from an automotive supply chain", *Decision Support Systems*, Vol. 142, p. 113452, 2021, doi: 10.1016/j.dss.2020.113452.

[10] 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.

[11] A. K. Ard, A. Bekrar, A. A. E. Cadi, and Y. Sallez, "Artificial intelligence for forecasting in supply chain management: a case study of White Sugar consumption rate in Thailand", *IFAC-PapersOnLine*, Vol. 52, No. 13, pp. 725–730, 2019, doi: 10.1016/j.ifacol.2019.11.201.

[12] 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.

[13] R. J. Hyndman and G. Athanasopoulos, *Forecasting: principles and practice*, OTexts, 2018.

[14] 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.

[15] "Tutorial: Podstawy stosowania Pythona do prognozowania szeregów czasowych", *Margo*. https://www.margo-group.com/pl/aktualnosci/tutorial-podstawy-stosowania-pythona-do-prognozowania-szeregow-czasowych/ (accessed Nov. 05, 2021).

[16] H. Min, "Artificial intelligence in supply chain management: theory and applications", *International Journal of Logistics Research and Applications*, Vol. 13, No. 1, pp. 13–39, 2010, doi: 10.1080/13675560902736537.

[17] I. Minis and N. Ampazis, "Applications of Neural Networks in Supply Chain Management", *Handbook of Research on Nature-Inspired Computing for Economics and Management*, pp. 589-607, 2007, doi:10.4018/978-1-59140-984-7.ch039.

[18] S. G. Timme and C. W. Timme, "THE REAL COST OF HOLDING INVENTORY.", *undefined*, 2003, Accessed: Jul. 08, 2021. [Online]. Available: https://www.semanticscholar.org/paper/THE-REAL-COST-OF-HOLDING-INVENTORY.-Timme-Williams-Timme/93e1f9d42b9f68fe8e37d4c0d4ebc2c1387e03e5

[19] R. Carbonneau, K. Laframboise, and R. Vahidov, "Application of machine learning techniques for supply chain demand forecasting", *European Journal of Operational Research*, Vol. 184, No. 3, pp. 1140–1154, 2008, doi: 10.1016/j.ejor.2006.12.004.

[20] G. B. Huang, Q. Y. Zhu, and C. K. Siew, "Extreme learning machine: Theory and applications", *Neurocomputing*, Vol. 70, No. 1,

pp. 489–501, 2006, doi: 10.1016/j.neucom.2005.12.126.

[21] T. Szul, K. Necka, and T. G. Mathia, "Neural methods comparison for prediction of heating energy based on few hundreds enhanced buildings in four season's climate", *Energies*, Vol. 13, No. 20, 2020, doi: 10.3390/en13205453.

[22] C. U. Vila, A. C. Z. D. Souza, J. W. M. Lima, and P. P. Balestrassi, "Electricity demand and spot price forecasting using evolutionary computation combined with chaotic nonlinear dynamic model", *International Journal of Electrical Power & Energy Systems*, Vol. 32, No. 2, pp. 108–116, 2010, doi: https://doi.org/10.1016/j.ijepes.2009.06.018.

[23] C. Catal, A. Fenerci, B. Ozdemir, and O. Gulmez, "Improvement of Demand Forecasting Models with Special Days", *Procedia Computer Science*, Vol. 59, pp. 262–267, 2015, doi: https://doi.org/10.1016/j.procs.2015.07.554.

[24] R. Hribar, P. Potočnik, J. Šilc, and G. Papa, "A comparison of models for forecasting the residential natural gas demand of an urban area", *Energy*, Vol. 167, No. C, pp. 511–522, 2019.

[25] 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.

[26] J. Dabounou, "Deep Learning : Les réseaux de neurones récurrents (RNN)", Aug. 25, 2021. https://datavalue-consulting.com/deep-learning-reseaux-neurones-recurrents-rnn/ (accessed Sep. 20, 2021).

[27] 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.

[28] 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, May 2008. doi: 10.1109/ICCCE.2008.4580831.

[29] A. Aktepe, E. Yanık, and S. Ersöz, "Demand forecasting application with regression and artificial intelligence methods in a construction machinery company", *Journal of Intelligent Manufacturing*, Vol. 32, pp. 1587–1604, 2021, doi: 10.1007/s10845-021-01737-8.

[30] A. Rahman, V. Srikumar, and A. D. Smith, "Predicting electricity consumption for commercial and residential buildings using deep recurrent neural networks", *Applied Energy*, Vol. 212, No. C, pp. 372–385, 2018.

[31] 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).

[32] 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.

[33] E. Dave, A. Leonardo, M. Jeanice, and N. Hanafiah, "Forecasting Indonesia Exports using a Hybrid Model ARIMA-LSTM", *Procedia Computer Science*, Vol. 179, pp. 480–487, 2021, doi: 10.1016/j.procs.2021.01.031.

[34] Y. Wu, M. Yuan, S. Dong, L. Lin, and Y. Liu, "Remaining useful life estimation of engineered systems using vanilla LSTM neural networks", *Neurocomputing*, Vol. 275, pp. 167–179, 2018, doi: 10.1016/j.neucom.2017.05.063.

[35] J. Zheng, C. Xu, Z. Zhang, and X. Li, "Electric load forecasting in smart grids using Long-Short-Term-Memory based Recurrent Neural Network", In: *Proc. of 2017 51st Annual Conference on Information Sciences and Systems (CISS)*, pp. 1–6, Mar. 2017, doi: 10.1109/CISS.2017.7926112.

[36] 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

[37] N. B. Syam and A. Sharma, "Waiting for a sales renaissance in the fourth industrial revolution: Machine learning and artificial intelligence in sales research and practice", *Industrial Marketing Management*, Vol. 69, 135–146, 2018, doi: 10.1016/J.INDMARMAN.2017.12.019.

[38] S. Jaipuria and S. S. Mahapatra, "An improved demand forecasting method to reduce bullwhip effect in supply chains", *Expert Systems with Applications*, Vol. 41, No. 5, pp. 2395–2408, 2014, doi: 10.1016/j.eswa.2013.09.038.

[39] J. Feizabadi, "Machine learning demand forecasting and supply chain performance", *International Journal of Logistics Research and Applications*, pp. 1–24, 2020, doi: 10.1080/13675567.2020.1803246.

[40] A. E. Filali, E. H. Ben Lahmer, and S. E. Filali, "Exploring applications of Machine Learning for supply chain management", In: Proc. of *2021 Third International Conference on Transportation and Smart Technologies (TST)*, pp. 46-52, 2021, doi: 10.1109/TST52996.2021.00015.

[41] 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*, Vol. 11, pp. 1463–1482, 2020, doi: 10.1007/s13042-019-01050-0.

[42] 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.

[43] A. A. Nasser, M. Z. Rashad, and S. E. Hussein, "A Two-Layer Water Demand Prediction System in Urban Areas Based on Micro-Services and LSTM Neural Networks", *IEEE Access*, Vol. 8, pp. 147647–147661, 2020, doi: 10.1109/ACCESS.2020.3015655.

[44] 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.

[45] J. Wang and A. Duggasani, "Forecasting hotel reservations with long short-term memory-based recurrent neural networks", *Int. J, Data Sci. Anal.*, Vol. 9, No. 1, pp. 77–94, 2020, doi: 10.1007/s41060-018-0162-6.

[46] J. F. L. D. Oliveira and T. B. Ludermir, "A hybrid evolutionary decomposition system for time series forecasting", *Neurocomputing*, Vol. 180, pp. 27–34, 2016, doi: 10.1016/j.neucom.2015.07.113.

[47] H. C. W. Lau, G. T. S. Ho, and Y. Zhao, "A demand forecast model using a combination of surrogate data analysis and optimal neural network approach", *Decision Support Systems*, Vol. 54, No. 3, pp. 1404–1416, 2013, doi: 10.1016/j.dss.2012.12.008.

[48] J. Shahrabi, E. Hadavandi, and S. Asadi, "Developing a hybrid intelligent model for forecasting problems: Case study of tourism demand time series", *Knowledge-Based Systems*, Vol. 43, pp. 112–122, 2013, doi: 10.1016/j.knosys.2013.01.014.

[49] 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.

[50] H. S. Khargharia, R. Santana, S. Shakya, R. Ainslie, and G. Owusu, "Investigating RNNs for vehicle volume forecasting in service stations", In: *Proc. of 2020 IEEE Symposium Series on Computational INTELLIGENCE (SSCI)*, 2020, pp. 2625–2632.

[51] D. Solyali, "A comparative analysis of machine learning approaches for short-/long-term electricity load forecasting in Cyprus", *Sustainability (Switzerland)*, Vol. 12, No. 9, 2020, doi: 10.3390/SU12093612.

[52] K. Uyar, Ü. Ilhan, and A. İlhan, "Long Term Dry Cargo Freight Rates Forecasting by Using Recurrent Fuzzy Neural Networks", *Procedia Computer Science*, Vol. 102, pp. 642–647, 2016, doi: 10.1016/j.procs.2016.09.455.

[53] S. S. Shwartz and S. B. David, *Understanding Machine Learning: From Theory to Algorithms*. Cambridge: Cambridge University Press, 2014. doi: 10.1017/CBO9781107298019.

[54] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, MIT Press, 2016.

[55] A. Graves, *Supervised Sequence Labelling with Recurrent Neural Networks*, Vol. 385. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012. doi: 10.1007/978-3-642-24797-2.

[56] 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.

[57] S. Hochreiter, Y. Bengio, P. Frasconi, and J. Schmidhuber, "Gradient Flow in Recurrent Nets: The Difficulty of Learning Long-Term Dependencies" *Wiley-IEEE Press*. pp. 237–243.2001, doi: 10.1109/9780470544037.ch14.

[58] T. Fischer and C. Krauss, "Deep learning with long short-term memory networks for financial market predictions", *European Journal of Operational Research*, Vol. 270, No. 2, pp. 654–669, 2018, doi: 10.1016/j.ejor.2017.11.054.

[59] H. Liao, G. Pundak, O. Siohan, M. Carroll, N. Coccaro, Q. Jiang, T. Sainath, A. Senior, F. Beaufays, and M. Bacchiani, "Large vocabulary automatic speech recognition for children", In: *Proc. of Interspeech*, 2015.

[60] A. Ghatak, "Introduction to Machine Learning", in *Machine Learning with R*, A. Ghatak, Ed. Singapore: Springer, pp. 57–78, 2017, doi: 10.1007/978-981-10-6808-9_3.

[61] G. E. P. Box, G. M. Jenkins, G. C. Reinsel, and G. M. Ljung, "Time Series Analysis: Forecasting and Control, 5th Edition", *Journal*

of Time Series Analysis, Vol. 37, No. 5 p. 709-711, 2016.

[62] F. Martínez, M. P. Frías, M. D. P. Godoy, and A. J. Rivera, "Dealing with seasonality by narrowing the training set in time series forecasting with kNN", *Expert Systems with Applications*, Vol. 103, pp. 38–48, 2018, doi: 10.1016/j.eswa.2018.03.005.

[63] 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.