OPERATIONS AND SUPPLY CHAIN MANAGEMENT
Vol. 16, No. 2, 2023, pp. 265 - 278
ISSN 1979-3561 | EISSN 2759-9363

Improving Lead Time Forecasting and Anomaly Detection for Automotive Spare Parts with A Combined CNN-LSTM Approach

Asmae Amellal
Laboratory of Modeling, Optimization of Industrial and Logistics Systems
Mosil, Ensa Tétouan, Morocco
Email: asmae.amellal57@gmail.com (*Corresponding Author*)

Issam Amellal
Laboratory of Modeling, Optimization of Industrial and Logistics Systems
Mosil, Ensa Berrchid, Morocco
Email: amellal.issam@gmail.com

Hamid Seghiouer
Laboratory of Modeling, Optimization of Industrial and Logistics Systems
Mosil, Ensa Tétouan, Morocco
Email: hseghiouer@gmail.com

Mohammed Rida Ech-Charrat
Laboratory of Modeling, Optimization of Industrial and Logistics Systems
Mosil, Ensa Tétouan, Morocco
Email: charrat.mohammed@uae.ac.ma

ABSTRACT

This paper presents a solution to a challenge faced in the supply chain management of a spare parts distributor with a dispersed global supply network and local distribution network in Morocco. The problem is a lack of accurate lead time information, leading to difficulties in meeting customer demand. The proposed solution is a framework using an LSTM (Long Short Term Memory) model for lead time forecasting and anomaly detection. The framework combines CNN (Convolution neural network) -Bidirectional LSTM model for forecasting and an LSTM autoencoder with One-Class Support Vector Machine for anomaly detection. The data was obtained from a legal ERP system of a major automotive distributor in Morocco. The results show that the framework is effective in overcoming the lead time information issue, and the relevance of the methods used has been verified via precise performance indicators, such as the RMSE (Root of the mean of the square of the errors) testifying to the accuracy of the results and also through comparison with other models.

Keywords : convolution neural network, anomaly detection, forecasting, long-term short memory, one class support vector machine, supply chain management

1. INTRODUCTION

Sustainability, resilience, agility, effectiveness, and efficiency are the characteristics of the new SCM (Supply chain management), which must be maximized to meet the new demands of globalization, including the relocation of production (Pan *et al., 2021*). Meeting the various challenges and being up to date when managing a supply chain must be done by taking into account the major issues dictated by the problem to be dealt with and the SCM perspective. Mentzer *et al.* (2001) demonstrated that there was three perspectives to observe the SCM : The management philosophy which consider the entire process of the supply chain from the supplier to the end customer as being a single entity and define the main strategic lines relating to the systemic approach to the overview of the supply chain from the supplier to the end customer, to the overall orientation of the management of the various internal and external stakeholders , the implementation of the management philosophy which determines the practices and methods to be adopted to fit into the predetermined philosophy, such methods have been put forward by many authors through the literature (e.g. mutually sharing information (Ellram and Cooper 1990; Cooper *et al.*, 1997) , and finally the management process-oriented approach which aims to be more pragmatic by focusing on the supply chain in its basic function, which is to manage flows from the supplier to the end customer, in other words, from this perspective, the SCM includes all the processes involved in the flow of goods and services, from the initial sourcing of materials and components to the delivery of the final product to the customer. This includes managing production flows, information flows, and financial flows at every stage of the supply chain. This encompasses activities such as inventory management, logistics, and distribution, as well as the coordination and communication with suppliers, manufacturers, distributors, and customers, this approach must also put a particular focus on the customer according to several authors (Lambert *et al., 1998* ; Cooper *et al., 1997*).

Decision-making is a crucial aspect of the management process-oriented approach, as it involves using the information to make decisions about activities related to supply, production, inventory, and distribution. Supply chain management tools, such as supply chain management systems and data analysis tools, can help make decisions using real-time information and historical data, taking into account costs, deadlines, quality, and safety.

In the case of the supply chain relating to the sale of automotive spare parts, the need is simple and implies the ability to respond to a certain request within reasonable and precise deadlines in order to maintain a good level of customer satisfaction. To achieve this, the supply chain must be carefully managed and monitored, with a focus on forecasting, inventory management, and logistics. This includes working closely with manufacturers to ensure that spare parts are produced in a timely and efficient manner, and with distributors and wholesalers to ensure that spare parts are delivered to the right place at the right time. In addition, retailers and repair shops must also be able to access accurate and up-to-date information on stock levels and delivery times, in order to be able to provide customers with the spare parts they need. Overall, the automotive spare parts sales supply chain requires a holistic approach that takes into account the specificities of the sector and prioritizes the treatment of critical issues in an adequate manner to ensure customer satisfaction and business success. Among critical issues, forcasting have always been an essential tool for a successful SCM (Boone *et al., 2019*), and anomaly detection takes its importance from the fact that data is increasingly useful for companies whether for prediction or any other analysis and therefore unreliable information can have direct or indirect negative repercussions and decrease the company's ability to make efficient and productive decision-making and cause direct financial and reputation loss of the company (Glaser *et al., 2022*).

In recent years, e-commerce has also become an increasingly important aspect of the automotive spare parts sales supply chain, with many retailers and repair shops now offering online sales and home delivery services. This highlights the importance of accurate forecasting, inventory management, and logistics in the automotive spare parts sales supply chain. To meet these challenges, the supply chain must be able to respond quickly and efficiently to customer demands and also can anticipate and adapt to changes in demand. This requires a high level of coordination and communication between all the players in the supply chain, as well as the use of advanced technologies and tools to optimize and streamline processes. Additionally, in the context of e-commerce, a strong and efficient online sales platform is vital to ensure a seamless customer experience and to be able to deliver accurate and reliable lead times to customers. This includes integrating real-time inventory management systems, order tracking, and logistics management to ensure that orders are fulfilled on time and to the customer's satisfaction. Overall, the automotive spare parts sales supply chain faces new challenges that require a highly efficient, precise, and flexible approach. This can be achieved through the implementation of advanced technologies, efficient processes, and effective communication and coordination between all players in the supply chain.

Lead time is a very important subject, and it is even more so when it is about online sales, this task is the one we tried to address in this paper by dealing with both lead time forecasting as well as anomaly detection issues based on the same dataset obtained via the legal ERP system of a major automotive distributor in Morocco who wishes to set up a digital platform for the sale of spare parts. The aforesaid online sales project encountered the difficulty of being able to communicate precise deadlines to customers through the digital platform knowing that the parts come from different manufacturers located in different locations around the world. In order to deal with this subject, we used artificial intelligence models. We have chosen to work in an LSTM framework by applying a CNN-BiLSTM model for lead time forecasting and opting for an autoencoder LSTM with OCSVM for anomaly detection. The contribution of this work lies firstly in the fact of proposing a theoretical framework to both solve the problem of anomaly detection as well as the question of forecasting, based on the same dataset from the real field of automobile distribution in Morocco, secondly, the chosen models are applied to the resolution of the question of lead time little addressed in the literature and to which these models have never been applied, and finally, the relevance of the models and the framework proposed are evaluated through the comparison with other more traditional machine learning models. The rest of this paper is structured as follows: Section 2 gives an overview of the related work, Section 3 defines the way employed to collect the required data from the automotive distribution in Morocco as well as the background of the ANN (artificial neural network) models utilized in this research whereas Section 4 describes the proposed models and presents how they were applied to the case, Section 5 discusses the results obtained for both the forecasting and anomaly detection of the lead time issue. Finally, Section 6 presents a discussion area where the contributions of our work are explained, its limitations are detailed, and the conclusions and future work are presented.

2. RELATED WORK

Forecasting and anomaly detection have separately been widely discussed in the literature and in particular through the deployment of machine learning techniques (Nam *et al., 2020*; Lu *et al.*, 2020; Loureiro *et al.*, 2018; Fildes *et al.*, 2022; Kerdprasop *et al.*, 2019; Tran *et al.*, 2019)

Anomaly detection is a subject that has often been dealt with using machine learning given its importance and cross-functionality for different fields involving data analysis. The use of machine learning implies that the relevance of the results depends on the power of the applied model. Methods like isolation forest (Lesouple *et al.*, 2021; Ding and Fei, 2013) Kernel density estimation (Latecki *et al.*, 2007; Hu *et al.*, 2018; Zhang *et al.*, 2018) are limited when it comes to large datasets. Deep learning methods like RNN (Ullah and Mahmoud, 2022; Nanduri and Sherry, 2016) or CNN (Kwon *et al.*, 2018; Kim *et al.*,2018) a good and powerful options for anomaly detection but without taking into account the previous behavior of the data according to a given characteristic. And this is where LSTM finds all its interest because equipped with a memory, it is able to give very relevant predictions (Hochreiter and Schmidhuber,1997; Greff *et al.,* 2016). LSTM anomaly detection works by first predicting a sequence from past samples and then determining the anomaly by examining the prediction error, i.e. an anomaly stands out and its value is independent of the previous values (Chandola *et al.,* 2009). An interesting use of LSTM for anomaly detection is its combination with autoencoder, the idea behind is to use the LSTM for the encoder as well as for the decoder. During the forward propagation, The encoding operation aims to produce a compressed form of the input while the decoding one has the objective to reconstruct the obtained input which is then compared to the real one, the difference between the two inputs represents the error. The back-propagation is then operated to readjust the weights. In the case of anomaly detection, the error mentioned above is the basis, as soon as it exceeds a certain threshold determined beforehand, an anomaly can be indicated.

OCSVM is a classification algorithm. It constitutes 10% of the usage rate in tackling forecasting problems (Aamer *et al*., 2020). Additionally, it has demonstrated its effectiveness in anomaly detection across numerous examples in the literature. Nevertheless, despite its notorious efficiency, OCSVM has some limitations, especially concerning gigantic and multi-dimensional databases due to the complexity of optimization required for this kind of data set. As mentioned above, LSTM autoencoder measures a certain difference from which an anomaly is determined. For this, there must necessarily be a threshold to refer to determine whether or not the aforesaid difference should indicate an anomaly and the choice of this threshold must not be arbitrary. Combining LSTM autoencoder with OCSVM could bring an answer to this threshold issue as in (Nguyen *et al.,* 2021) where the idea was to extract crucial information from the data by using LSTM autoencoder and then detect the anomalies by using the OCSVM.

In the same way as anomaly detection, forecasting has benefited greatly from advances in machine learning which provides effective solutions adapted to non-linear and voluminous data. The literature presents many machine learning methods that have been deployed for forecasting in all disciplines, the simple ones as random forest, lasso, RNN, LSTM, and BiLSTM (Dudek, 2015; Roy *et al.,* 2015; Balluff *et al.,* 2020; Siami-Namini *et al*., 2019) as well as the hybrid ones like RNN-LSTM (Chandriah and Naraganahalli., 2021), CNN-RNN, CNN-LSTM (Lu *et al.*, 2020; Vidal and Kristjanpoller, 2020). The relevance of the used model is each time assessed through general indicators and also through comparison with other simple or hybrid models. CNN-BiLSTM is a hybrid model that has often been used for forecasting, (Miao *et al*., 2021) used a CNN-BiLSTM model with Bayesian Optimization and Attention Mechanism for Short-term power load forecasting, the attention mechanism serves to have a particular concentration on the most important output and the Bayesian Optimization is used to tune the hyperparameters of the model, The method used to evaluate the performance of the model is the one mentioned above, namely the comparison with the results obtained via other models. (Chen *et al.*, 2022) used CNN-BiLSTM method based on feature selection for Short-Term Wind Power Forecasting, the idea was to optimize the feature parameters correlation based on effective feature screening of multidimensional feature datasets and then weighting the input data regarding the feature correlation before applying the CNN-BiLSTM model. CNN-BiLSTM model was used for stock price forecasting in many articles recently, sometimes by integrating an attention mechanism (Lu *et al.,* 2021; Chen *et al.*, 2021; Nourbakhsh and Habibi, 2022) or by applying a simple CNN-BiLSTM model (Wang *et al*., 2021), and each time, the relevance of the model was verified through comparison with other models.

Forecasting and anomaly detection, regardless of the method used, has been treated for several areas in general, and for the supply chain in particular. Concerning the supply chain, price, and stocks issues have been largely addressed. However, other supply chain issues are just as important as price and stock prediction issues, notably the lead time issue which has been less addressed by the literature and especially with regard to the use of machine learning. Recent articles have nevertheless dealt with this task (Babai *et al.,* 2022) tried to estimate the variance of the lead-time demand forecast error under stochastic lead-times by using the classical statistics model ARMA (autoregressive moving average) framework which works with the simple principle of forecasting using the previous values of the forecast variable (Auto-Regressive model), and using the mean plus the error terms (Moving Average model). The idea was to present three strategies for forecasting under the same framework and then compare them to choose the best one. However, the results obtained have not been compared to other methods, notably those of machine learning known to offer more precision, and therefore the relevance of the strategies presented has not been proven outside the ARMA framework. In Meisheri *et al*. (2022) authors opted for deep reinforcement learning (deep RL) to address the lead time issue for retail businesses with uncertain demand, the choice of the proposed approach was justified by the need to simultaneously manage stocks of a large number of products under realistic constraints, and by the need for effective management of the multi-period constraints resulting from different lead times of different products, the performances of the proposed RL approaches was compared to the baseline heuristics by experimenting on scales between 100 and 220 products, but has not been compared to other machine learning models to ensure that there is no oversizing in terms of the cost of the model chosen compared to the performance obtained.

3. RESEARCH METHODOLOGY

The research methodology is described in **Figure 1**.

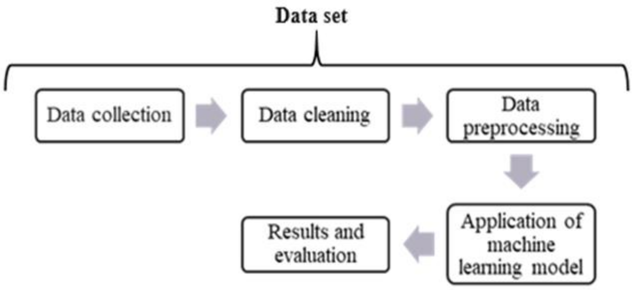

Figure 1. Analysis framework

3.1 Data Set

Data Collection

The data source is a Moroccan company that specializes in the importation and distribution of automobile spare parts. The company markets various automobile brands from different manufacturers and operates across multiple countries, which leads to a complex logistics process with varying laws and policies. The high number of references to manage and the volatile demand for certain parts make the process challenging. The data covers the period from January 2019 to August 2022. To address the challenges related to data quality, completeness, and accuracy, we took several measures. Firstly, we worked closely with the automobile company to obtain the necessary data and ensure that it was accurate and complete. This involved collaborating with the company's IT department to obtain the data from the ERP system. Moreover, we acknowledged the importance of data privacy and confidentiality in our study. As the data we were working with involved sensitive and proprietary information, we took measures to ensure that the data was anonymized and secured. We removed any personally identifiable information from the dataset, such as customer names or addresses, to protect the privacy of individuals. By taking these measures, we aimed to ensure that the data was reliable and suitable for analysis, while also protecting the privacy of individuals and complying with applicable data protection regulations.

Data Cleaning

We performed rigorous data cleaning and quality control procedures on our dataset, which consisted of 51,092 samples and 15 features. We conducted exploratory data analysis to detect any data gaps or inconsistencies and took appropriate steps to address these issues. This included checking for duplicate data, identifying and correcting missing or incomplete data, properly formatting and coding data, and verifying data consistency and accuracy. For instance, negative values present in the original dataset due to an internal process were replaced with zeros.

Data Preprocessing

The analysis of the data requires pre-processing before applying machine learning models to improve the results. We acknowledged the possibility of data biases, particularly if certain types of spare parts or suppliers were overrepresented in the dataset. Thus, we carefully examined the data to identify any potential sources of bias. To address this, we examined the distribution of the data by analyzing the frequencies and patterns of the data values in each variable or feature of the dataset using visualization techniques, such as density plots or histograms. We also checked for any over- or under-represented groups or variables in the dataset. Additionally, we used a correlation matrix to determine the most relevant features, resulting in us keeping only 8 features (summarized in **Table 1**). **Table 1** shows the values of the correlation coefficients for each of the 8 features with the lead time.

Table 1. The correlation coefficient values with lead time

Inputs	Correlation value
Supplier	0.2
Reference	0.03
Designation	-0.002
Available quantity	-0.03
Sale's price	-0.005
Age	0.02
Selling location	-0.004
Lead time	1

After identifying potential biases and selecting the relevant features, we used additional data preprocessing techniques to prepare the data for machine learning models. These techniques included:

- Data encoding of categorical variables to convert them into numerical values so that they can be used by machine learning models. This process can involve converting text data into numerical data by using techniques such as one-hot encoding, label encoding, or binary encoding. The result will be numerical variables with either 32 or 64 bits integers.
- Normalization of data to bring it to a common scale. Min-Max normalization, also known as feature scaling, is a technique used to normalize the data by transforming it into a range of 0 to 1. This is achieved by subtracting the minimum value of the data from each observation and dividing it by the difference between the maximum and minimum values of the data. The formula for Min-Max normalization is as follow :
-

$$x_{\min} = \frac{(x - x_{\min})}{(x_{\max} - x_{\min})} \quad (1)$$

- Sampling to split the data into a training sample and a test sample. The data is split into two parts: the training sample and the test sample. The training sample is used to train the machine learning models, while the test sample is used to evaluate their performance. In this case, the training sample consists of 40,873 samples, and the test sample consists of 10,219 samples.

3.2 Background

In this section, we briefly overview the deep learning algorithms RNN, LSTM&BiLSTM, CNN, autoencoder, and OCSVM which were used to set up the proposed algorithm for forecasting and anomaly detection.

3.2.1 Recurrent Neural Network (RNN)

A Recurrent Neural Network (RNN) is a type of artificial neural network. Its field of use is very broad and its ability to recognize recurrences, as its name suggests, makes it a very good ally when it comes to recognizing data sequences such as for machine translation, automatic recognition of speech, automatic writing proposal, and many other applications including the analysis of time series which is of capital importance. RNN, like other neural network algorithms, is intended to make predictions as well as classifications. Basically, RNN has the architecture of an MLP (Multilayer perceptron) to which loops have been added. (Manaswi, 2018). **Figure 2**. Represents A simple artificial neuron and Multilayered artificial neural network.

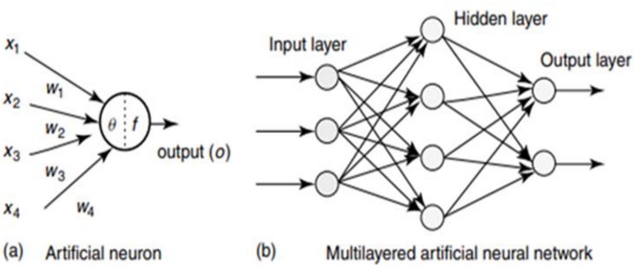

Figure 1 A simple artificial neuron and multilayered artificial neural network

(a) $(x_1,...x_4)$ are the inputs, $(w_1,...w_4)$ are the weights which can be modified so as to model synaptic learning, and f the activation function (which is often a sigmoid function) :

$$\Theta = \sum x_i w_i + b \; ; \; f(\Theta) = o \tag{2}$$

(b) A simple artificial neuron is called a node, a Multilayered artificial neural network is composed of several nodes in each layer working in a similar way to the simple artificial neuron. The outputs of each layer work as inputs for the next layer.

Figure 3 shows a generic structure of RNN network : The output at a given instant depends on the inputs at this same instant as well as at previous instants, this can be expressed in the following way:

$$h_t = f(h_{t-1}, x_t) \tag{3}$$

where ht refers to the output of a hidden layer of an RNN at the instant t, h_{t-1} the output at the instant (t-1), and xt the input at the instant (t).

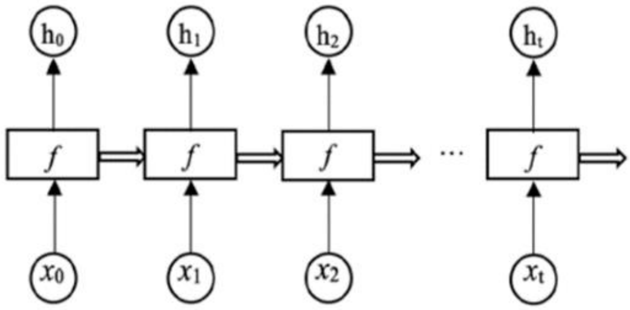

Figure 2. Recurrent layer over time steps

3.2.2 Long Short-Term Memory (LSTM) & Bidirectional LSTM

Long Short-Term Memory (LSTM) was proposed for the first time by Hochreiter&Shmidhuber in 1997 as a novel recurrent network architecture in conjunction with an appropriate gradient-based learning algorithm with the objective to overcome long-term dependencies problem (Hochreiter and Shmidhuber, 1997). Traditional RNNs networks present vanishing and exploding gradient problems, so the main idea of LSTM networks was to overcome those problems by adding the « memory » concept. **Figure 4**. Shows the modern representation of LSTM cells (Smagulova *et al.*, 2019).

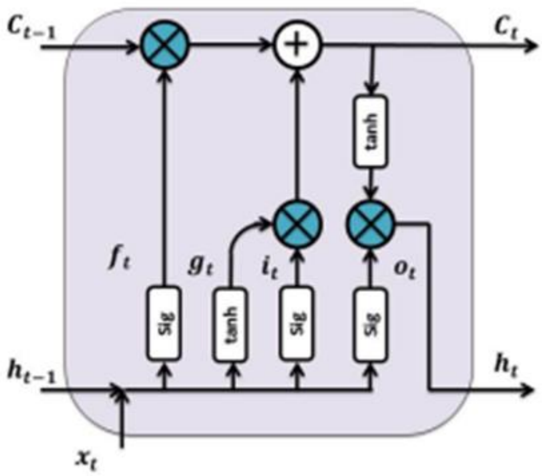

Figure 3 Modern representation of LSTM

The input of an LSTM unit is composed of three distinct vectors, the first, named input vector, is external to the LSTM unit and is injected into it at instant t (The vector x t in the diagram above), the two of them are generated by the LSTM unit at the previous instant (t-1) (In the diagram, they are the vector h_{t-1} indicating the hidden state, and the vector c_{t-1} representing cell state).

The cell c_t represents the long-term memory at time t, which, at time t, is updated thanks to information obtained from outside and those of the short-term memory at instant (t- 1) (x t-1 and h t-1 respectively). The gate (LSTM unit) controls the flow of information to contrast the vanishing gradient problem.

LSTM is built in such a way that it can work in one or more directions, so BiLSTM can be seen as the superposition of two unidirectional LSTMs with two opposite directions, one corresponding to the forward direction and the other to the backward direction. This provides the LSTM network the ability to use recent information as well as older ones at any time during its operation. **Figure 5**. Shows the unidirectionnel and bidirectionnel LSTM model (Smagulova *et al*., 2019).

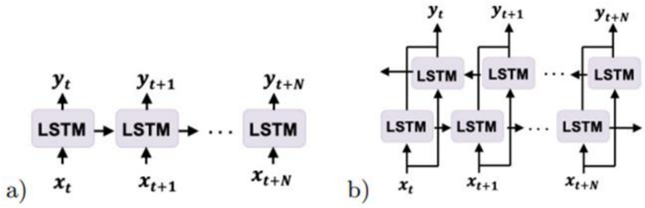

Figure 4 a) Unidirectional LSTM; b) BiLSTM

3.2.3 Convolution Neural Network (CNN)

Convolution Neural Network (CNN) is one of the most popular deep learning architectures. Mainly, CNN is used in image-related problems, Language Modeling, Sentiment Analysis, Language Translation, and more.

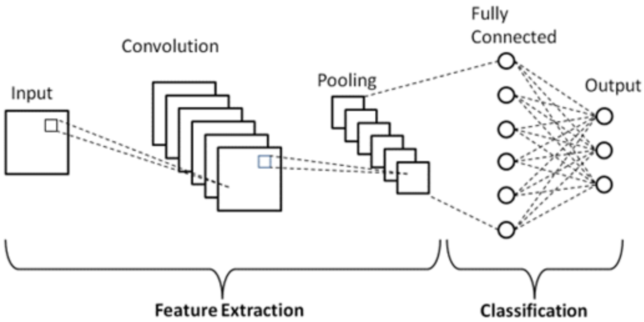

Figure 5 CNN Basic architechture

Figure 6 shows the basic CNN architecture. Height, width, and depth represent the dimensions of the input x of each layer in a CNN model. Given (α) the value of the height and the width which are equal, and (γ) the value of depth which refers to the channel numbers (For a black and white image for example, (γ) is equal to two). k represents the number of filters (called kernel) composing each convolutional layer. Like input x, k is three-dimensional (β, ϴ, and **Ω** the three dimensions) and the dimensions in question must satisfy the following conditions: ϴ < β and **Ω** < = γ. Local connections are handled by kernels and they have the same parameters, note b^k biases and W^k pods, so a number k of α_k feature maps are convolved with input after being emitted with size (β - ϴ -1). Equation (2) represents the computation performed by the convolution layer which is worked in two parts, a dot product between the input and the weights of the convolution layer, then applying an activation function :

$$f(W^k * x + b^k) = \alpha_k \quad (4)$$

Then the pooling is applied for each feature map, which represents the operation where features are extracted from the area undergoing the convolution process by the kernel (the extraction could be maximum or average), the area concerned by the pooling application is an adjacent one of the size (*k* x *k*) where *k* is the kernel size. The last step works like an ordinary neural network where the complete connection layers receive as inputs the functionalities resulting from the convolution operation. The final layer is in charge of determining the classification scores which represent the probabilistic value of each class (Alzubaidi *et al.*, 2021).

3.2.4 One-class support vector machine(OCSVM)

SVM (Support vector machine) is a type of supervised machine learning classification algorithm, the currently known version of SVM is the one that was adjusted by Vapnik and Corinna Cortes in 1995 (Cortes and Vapnik, 1995). Classification is the main objective of an SVM algorithm, it tries to separate each class using a hyperplane, the idea is to determine the optimal hyperplane in association with the SVM function coefficients. The goal is to maximize the margin between the linear line and the nearest positive and negative support vectors (The support vectors are the data points on the margin boundary).

One-class support vector machines (OCSVM) is an SVM type that considers only one class of data instead of two as for SVM. So, OCSVM is a one-class classification technique that also aims to find the maximum-margin hyperplane that best separates the training data from the origin (Schölkopf *et al.* 1999).

Consider $\{x_i\}(i = 1, \dots m)$ as a sample of points, with $x_i \in IR_f$. Given the function Φ: $IR_f \rightarrow K$ whose starting set IR_f represents the space of features while the set of results K is a space generated by the kernel k(x, y) and named space of kernels, the transformation is applied to each point x_i by k(x, y) = <Φ (x), Φ (y)>. OCSVM algorithm. In the case where it is impossible to perform such a separation (non-existent hyperplane), certain points are allowed in the margin by the deviation variables γi and at a cost controlled by the parameter ϴ ∈ [0, 1]. In general, ϴ provides an upper bound on the fraction of outliers (Schölkopf *et al.*, 2001). The OCSVM solves the quadratic program:

$$\min_{(V,\, \gamma i,\, \beta)} \frac{1}{2}\left\| v \right\|^2 + \frac{1}{\Theta n}\sum_{i=1}^{n}(\gamma i - \beta) \quad (5)$$

With:(v. Φ $(x_i)) \geq b- \gamma_i$, $\gamma_i \geq 0$, ∀i =1, n (6)

Where v refers to a vector perpendicular to the hyperplane, β is the distance to the origin, γi ≥ 0 are slack variables to deal with anomalous values that may be part of the dataset distribution, and ϴ ∈ (0, 1] the parameter in charge of controlling the trade-off between the number of examples in the training set mapped as positive by the decision function. The optimization problem in equation (3) is usually solved by its dual form:

$$\text{Minimise} : -\sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j K(x_i, x_j) \quad (7)$$

$$\text{With} : \sum_{i=1}^{n} \alpha_i = 1,\ 0 \leq \alpha_i \leq 1/vn \quad (8)$$

Where $K(x_i, x_j) = \Phi(x_i)^T \cdot \Phi(x_j)$ is the kernel function or so-called nonlinear projection function.

4. PROPOSED FRAMEWORK AND APPLICATION TO THE USE CASE

In this section, we will overview the models chosen and explain the relevance of this choice by considering our tasks namely lead time forecasting and anomaly detection. **Figure 7** Shows the proposed LSTM framework proposed to deal with aforesaid issues.

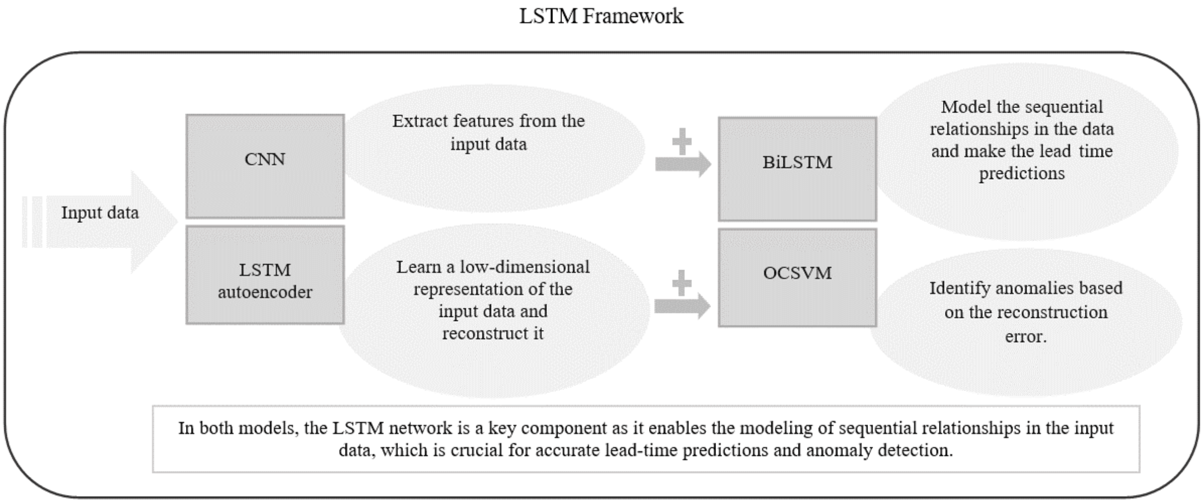

Figure 7. The proposed LSTM framework

4.1 CNN--BiLSTM for Lead Time Forecasting

4.1.1 Proposed Model

Proposed model is presented in **Figure 8**. The role of CNN is the efficient extraction of features. The idea behind is to use it to reduce the number of parameters between the connection layers, which represents an advantage that CNN has and that traditional neural networks do not have. Bidirectional LSTM acts as forward and backward LSTM networks for each training sequence, The BiLSTM block composed of two LSTM networks produces a single layer as output which ensures a complete connection from the initial input for each data frame. So, the BiLSTM is used to make a more accurate prediction model.

In our issue, Each of the two blocks, namely CNN and BiLSTM, plays a specific role, CNN starts by working on the data composing the input by extracting the features, then BiLSTM predicts the lead times of the parts using the extracted features recovered from the first block.

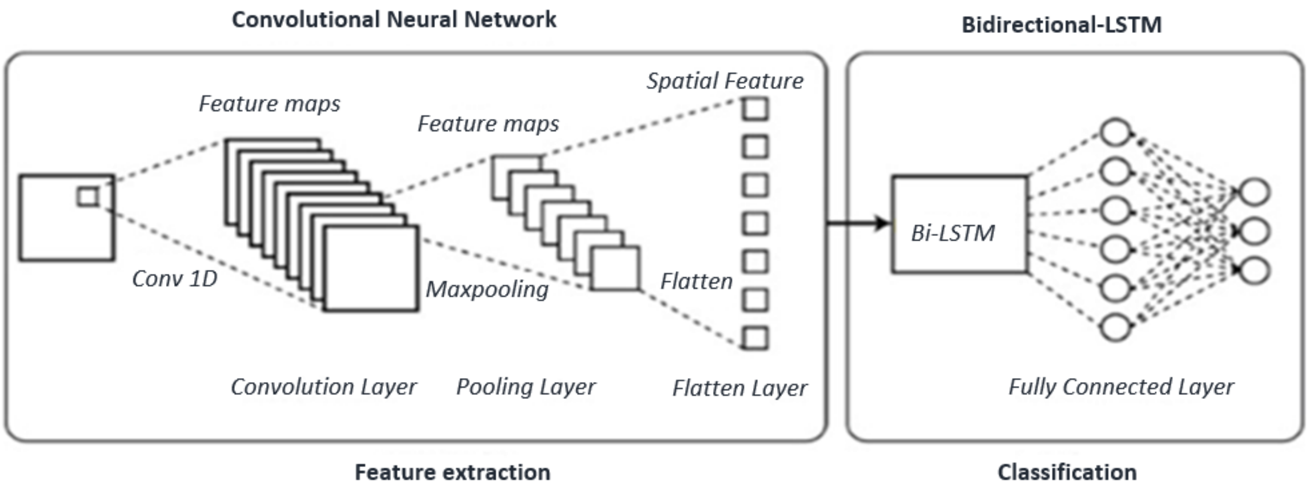

Figure 6 CNN-BiLSTM Basic

4.1.2 Experiment Setting and Implementation

Figure 1 illustrates the structure of the program and the parameter settings of the proposed method. In implementing the model, it's crucial to specify the parameters that will be used to evaluate it. The following metrics were used for evaluation: Mean Square Error (MSE), Root Mean Square Error (RMSE), and Mean Absolute Error (MAE).

Table 2. Criteria metrics

Metric	Definition	Formula
MAPE (Mean absolute percentage error)	The Measure of the difference between model-fitted values and observed data values.	$MAPE = \frac{100\%}{n} \sum \frac{\lvert(y_i - y_p)\rvert}{\lvert y_i \rvert}$
MAE (Mean of Absolute value of Errors)	The Measure of the errors between the values fitted by the model and the values of the observed data.	$MAE = \frac{\lvert(y_i - y_p)\rvert}{\lvert y_i \rvert}$
MSE (Mean of the Square of Errors)	The Measure of the errors between the values fitted by the model and the values of the observed data with a penalization of very large or outlying errors.	$MSE = \frac{\sum(y_i - y_p)^2}{n}$
RMSE (Root of the Mean of the Square of Errors)	It is the square root of the MSE, which means that it is measured in the same units as the response variable. In fact, it is used when it comes to evaluating the adequacy of a model to a dataset.	$MSE = \sqrt{\frac{\sum(y_i - y_p)^2}{n}}$

Table 2 provides a summary and explanation of these metrics, where y_i represents the actual value, y_p represents the predicted value, and n is the number of observations or rows. These metrics are commonly used to evaluate the performance of regression models and provide insight into the accuracy and reliability of the model's predictions.

The network structure was implemented using the Python programming language and the neural network framework used was Keras. The combination of Python and Keras provides a flexible and user-friendly environment for building and training deep learning models, making it well-suited for this task. **Figure 9** shows the diagram of the model construction.

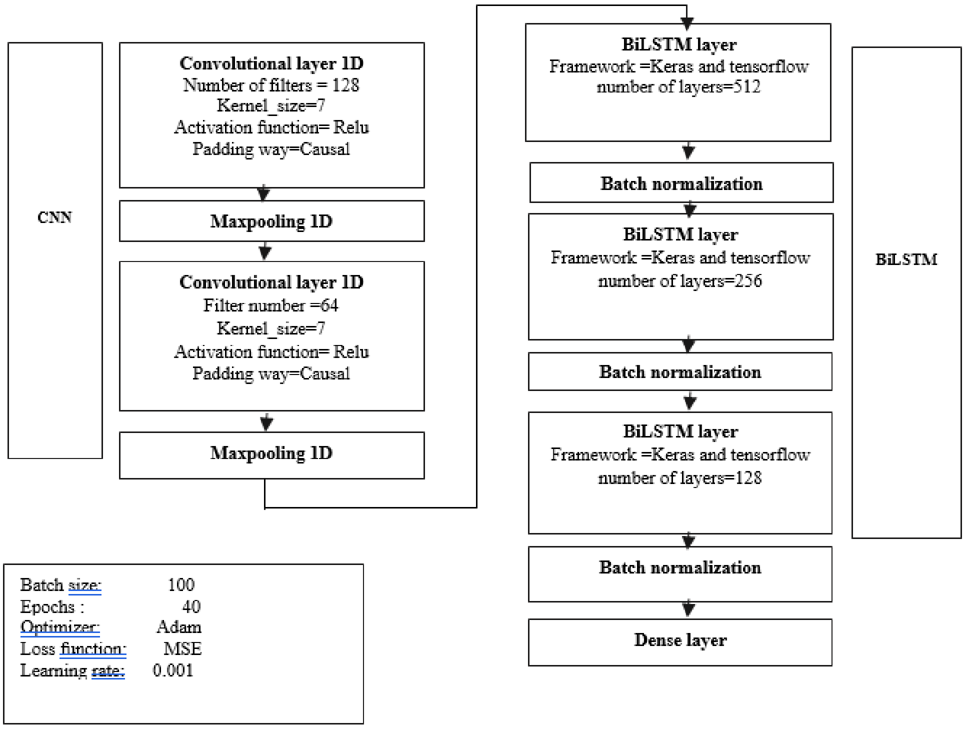

Figure 7. Program construction diagram (CNN-BiLSTM)

Where « Number of filters » are the number of output channels, « Kernel_size » defines the size of the convolution filter, «Relu » (Rectified Linear Unit) is the activation function used to transform the input signal which becomes the new input signal for the next layer, the « Padding » represents the extention of the data area by adding zeros as a way to avoid the information lost during the conventional process (Here since the CNN use is 1 dimension, the padding way is automatically " causal " which means the action to pads the front of the input layers with zeros), the « Maxpooling » represents the pooling operation where the maximum features are extracted from the area undergoing the convolution operation by the kernel, the « framework » of the BiLSTM model is keras and tensorflow libraries which gives a python interface, the « Number of layers » represents the number of hidden layers in the built model, the « Batch normalisation » is the usual normalization but which is applied only to a part of the data called « Batch » which results from the separation of data during the training (This operation speeds up the training operation through the possibility of using a higher learning rate), the « Batch_size » is the number of « Batch » gived by the above separation, « Epochs» refers to is the number of times the entire training operation will be performed, « Optimizer » is the mathematical function used to minimize the error function (the function used here is "Adam" which is an advanced version of the gradient decent method), and the « Learning rate » defines the value with which it is necessary to adjust the parameters of the model in order to better approach the minimum of the error function (This is a parameter that is determined beforehand and which represents the jump made at each descent of the gradient towards the minimum of the cost function).

4.2 LSTM Autoencoder for Anomaly Detection

4.2.1 Proposed Model

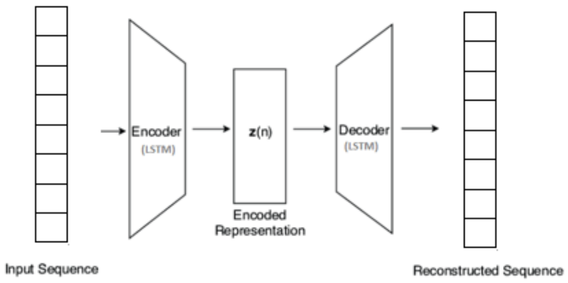

Figure 10. LSTM Autoencoder

Auto-encoders are generally used in unsupervised learning which consists of training the machine from non-annotated examples. In this case, the algorithm operates without first receiving example results (Sutskever *et al.*, 2014). An autoencoder consists of an encoder and a decoder. An encoder is a transformation performed via the function **E : X ->F** who, from a data space **X**, learns the main features and creates a new (encoded) data frame that is part of the feature space **F.** The decoder is also a transformation performed via the function **D : F → X** that has the objective to reconstruct the initial input using the internal representation.

The mission of the autoencoder is to compress the input into a code (Called « the latent-space representation »), with a lower dimension which is a “compression” of the input. The output is then reconstructed from this representation. This method allows us to keep intact the major information of the dataset. Autoencoders have been widely proposed in the literature such as convolutional autoencoders (Masci *et al.*, 2011), denoising autoencoders (Vincent *et al.*, 2011), contractive autoencoders (Rifai *et al.*, 2011) and LSTM autoencoder was used to handle with different issues, especially with anomaly detection (Nguyen *et al.*, 2021 ; Maleki *et al.*, 2021 ; Provotar *et al.*, 2019 ; Lindemann *et al.*, 2020) For our anomaly detection problem, we used LSTM cells to implement both the encoder and the decoder. Temporal dependencies between sequences are extracted using LSMT cells. To fix the anomaly detection issue, the approach applied is based on semi-supervised learning, which consists of using an autoencoder training data set free of anomalies. In other words, the autoencoder is a function Z_{AE} EoD which gives **:**

$$Z_{AE}x(n) = y(n) \tag{9}$$

Knowing that x(n) is a simple sequence, given (*m*) the sample length. The autoencoder is trained by minimizing the reconstruction error :

$$L = \frac{1}{2}\sum_{1}^{m} \|y - x\|^{2} \tag{10}$$

4.2.2 Experiment Setting and Implementation

To summarize the performance of our model, the following metrics were used :

$$Accuracy = \frac{TP+TN}{TP+FP+TN+FN} \tag{11}$$

$$Recall = \frac{TP}{TP+FN} \tag{12}$$

$$Precision = \frac{TP+TN}{TP+FP} \tag{13}$$

$$F - Score = \frac{2*Precision*Recall}{Precision+Recall} \tag{14}$$

Where TP (True Positive) represents the number of anomalies correctly recognized as anomalies, TN (True Negative) represents the number of normal events correctly recognized as normal, FP (False Positive) represents the number of normal events incorrectly recognized as anomalies, and FN (False Negative represents the number of anomalies incorrectly diagnosed as normal events. Accuracy determines how well the model predicts the output. Precision calculates, out of a total number of positive outputs (true or false), how many are really consistent with reality. As for the recall, it indicates the success rate of the model in predicting the positive outputs compared to the total number of positive classes. And finally F-Score is a metric used to determine both recall and precision.

Python programming language was used to implement network structure and Keras as the framework of neural networks. **Figure 11** illustrates the construction of the model. The aim is to extract the key features from the multivariate time series by using a Long Short-Term Memory (LSTM) autoencoder. This eliminates the dependency within the multivariate time series and results in independent error vectors (the difference between the input and the output of the trained LSTM autoencoder). Then, the One-Class Support Vector Machine (OCSVM) can define a hyperplane to distinguish between abnormal observations and normal samples based on these vectors. The use of LSTM autoencoder for feature extraction and OCSVM for anomaly detection provides a robust and effective solution for anomaly detection in multivariate time series data. This hybrid approach leverages the strengths of both methods to provide a more accurate and reliable result compared to using either method alone.

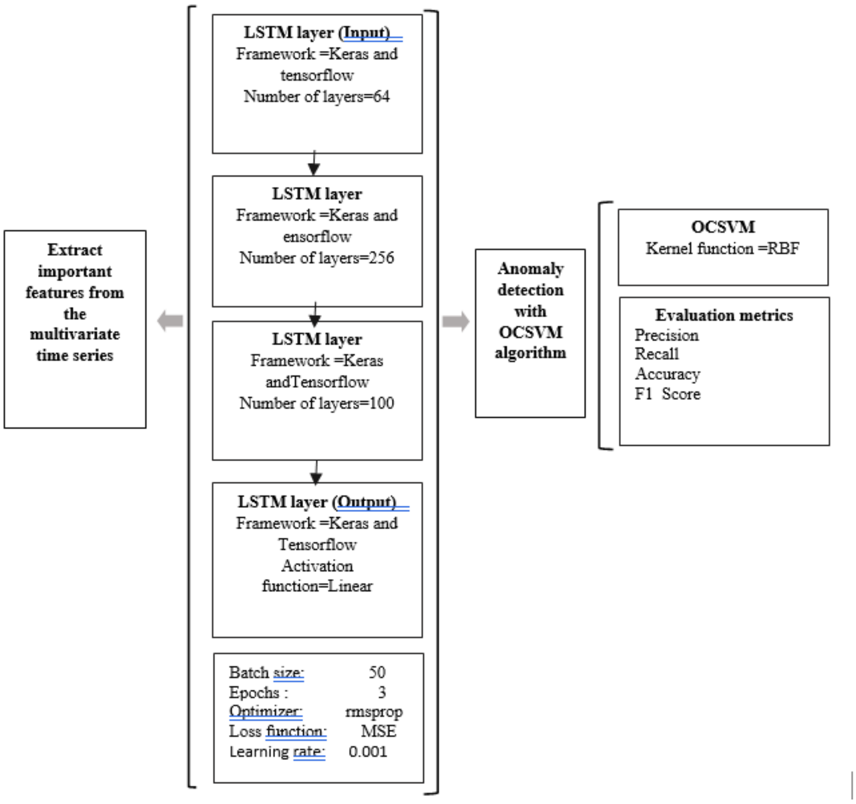

Figure 8 Program construction diagram (LSTM autoencoder-OCSVM)

5. RESULTS AND EVALUATION

For the experiment we used the data set described in section 3.1, the setting are those defined in sections 4.1-b and 4.2-b.

5.1 Lead Time Forecasting with CNN-BiLSTM

The hyperparameters used for the model, such as the number of layers and the learning rate, are specified in section 4.1-b. The computation was performed on a Jupyter Notebook platform equipped with an Intel(R) CORE(TM) i5 processor and 8 GB of RAM, GPU: Intel UHD Graphics, Operating System: Windows 10 Pro 64-bit. The training process took 31,058 seconds to complete.

Figure 12 provides a comparison between the predicted lead times and the actual lead times from the testing set. The corresponding line plots of train loss and validation loss are also included, allowing for a visual evaluation of the model's performance. The results are satisfactory, with the predicted and actual data being quite close, indicating the effectiveness of the proposed CNN-BiLSTM model.

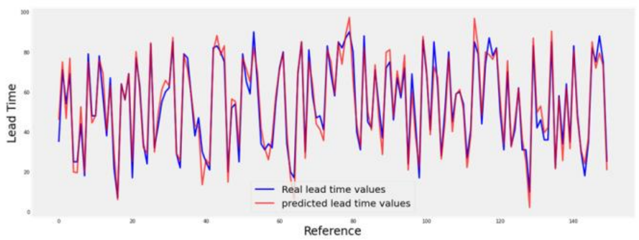

Figure 9 Real vs predicted lead time using BiLSTM-CNN

Also, errors on the training sets and the validation sets decrease significantly after a few epochs, which proves the effectiveness of the model CNN-BiLSTM regarding our task which is a time series forecasting one (**Figure 13**).

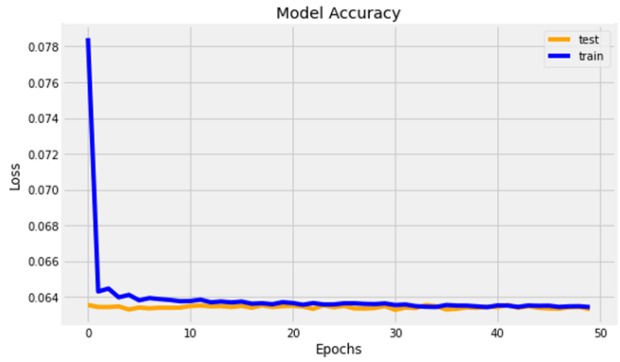

Figure 10 Line plot of train and validation loss from the CNN-BiLSTM model

To further demonstrate the effectiveness of the CNN-BiLSTM hybrid model for lead time prediction, a comparison was made with simple Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) models. The comparison was based on the Root Mean Squared Error (RMSE), Mean Squared Error (MSE), and Mean Absolute Error (MAE) metrics, as detailed in Table 2 of section 4.1-b. The results of this comparison are shown in **Table 3**.

Table 1 Criteria metrics results depending on model

	Simple RNN	LSTM	CNN-BiLSTM
MSE	0.0730	0.0723	0.0632
RMSE	0.25	0.24	0.21
MAE	0.21	0.21	0.19

Based on the results, it can be concluded that the proposed CNN-BiLSTM model outperforms simple Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) models in terms of the error values for the three metrics: Root Mean Squared Error (RMSE), Mean Squared Error (MSE), and Mean Absolute Error (MAE). This suggests that the combination of the CNN and BiLSTM networks provides a more accurate and robust solution for lead time forecasting, as compared to simple RNN and LSTM models. The use of the CNN for feature extraction and the BiLSTM for time-series prediction results in improved performance and a more effective solution for the problem at hand.

The obtained results demonstrate that our proposed lead time forecasting approach has the potential to provide significant benefits to spare parts suppliers and other organizations operating within the suply chain. By providing more accurate and reliable lead time forecasts, our approach can effectively reduce the risk of stockouts, minimize costs associated with overstocking, and improve transportation planning decision-making. This can result in improved operational efficiency, which can lead to a more sustainable and profitable business model for these organizations. In addition, improved lead time forecasting accuracy can also enhance customer satisfaction by ensuring timely delivery of products and reducing the potential for delays or disruptions. This can increase customer loyalty and repeat business, further contributing to the success of the organization.

Overall, our proposed lead time forecasting approach has the potential to bring significant benefits to spare parts distribution companies not only in Morocco but also beyond. By providing more accurate lead time forecasts, organizations can make more informed decisions, better manage their operations, and achieve a more resilient and responsive supply chain. Furthermore, our approach could encourage companies in Morocco to expand their online spare parts sales, resulting in increased efficiency and customer satisfaction.

5.2 LSTM Autoencoder for Anomaly Detection

For our case study, the LSTM autoencoder combined with OCSVM for classification brings better results than a simple LSTM autoencoder, we can see the comparison in **Table 4**. The metrics used to perform this comparison are those explained in setction 4.2-b. The results show that LSTM autoencoder performs better when combined with OCSVM than when it operates alone.

Table 2 Criteria metrics results depending on model

Model	Precision	Recall	Accuracy	F_Score
LSTM autoencoder	0.93	0.92	0.91	0.87
LSTM autoencoder with OCSVM	0.95	0.97	0.94	0.89

Also, the evolution of the loss function (Mean Square Error) curve for the test and validation sets shows that errors decrease significantly after a few epochs as shown in **Figure 14**.

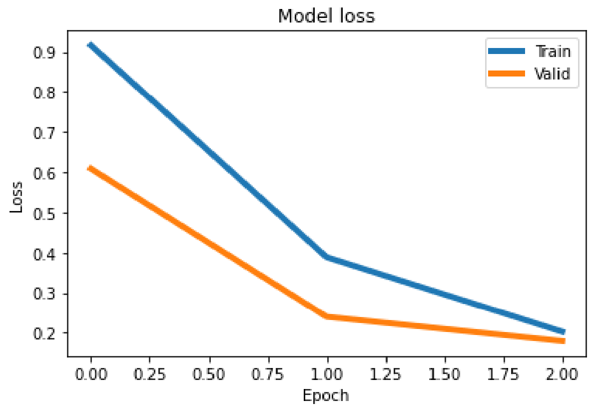

Figure 11 Line plot of train and validation loss from the LSTM autoencoder

Figure 15 compares the anomalies (represented as red dots) with the normal behavior relative to lead time. The lead time of each spare part is dependent on its manufacturer's production schedule, with some parts having a higher lead time due to delays in production. The anomaly detection capability of this approach is highly beneficial for optimizing supply operations. By detecting anomalies in real-time, it becomes possible to accurately determine the deadline for spare parts supplying, reducing the risk of error and providing customers with up-to-date information through the digital platform. This aligns with the initial goal of the project, which was to provide real-time information about supply deadlines to customers through the digital platform. The use of anomaly detection in this context can help ensure that this information is accurate and reliable, leading to improved customer satisfaction and better supply chain management. Our proposed approach can offer significant benefits for organizations engaged in online automotive spare parts sales in Morocco, which has seen relatively modest adoption rates due to various factors, including technical complexities. By leveraging our framework, these organizations can more effectively identify anomalies in their supply chain operations, such as unexpected changes in demand or disruptions in the distribution network, and take corrective actions to prevent future disruptions. This can lead to a more resilient and responsive supply chain that can adapt to unexpected events and maintain service levels even under adverse conditions.

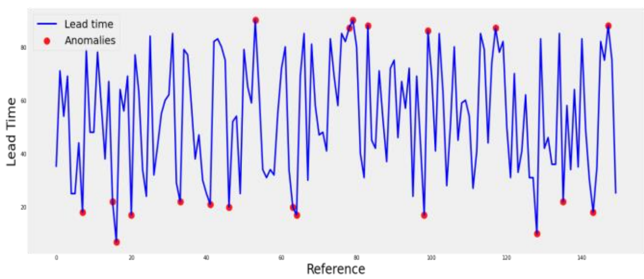

Figure 12 The anomaly detection for real data based on the LSTM autoencoder network and the OCSVM algorithm

6. DISCUSSION

6.1 Contributions

Our study presents a comprehensive theoretical framework for effectively addressing the challenges of forecasting and anomaly detection in the specific task of lead time. We evaluated the performance of our proposed models using real-world automobile distribution data from Morocco and demonstrated high performance for our CNN-BiLSTM model. Particularly when combined with the grid search technique for parameter tuning and our carefully designed data preprocessing steps, our model outperformed simple LSTM and RNN models applied to our task with the same dataset, as well as many other works that employed LSTM and RNN-based models for different tasks. Our results emphasize the significance of data preprocessing and parameter tuning, in addition to model complexity, for achieving better forecasting performance.

Our proposed forecasting model can be combined with classical techniques such as ARMA and ARMA-GARCH techniques to provide a more comprehensive understanding of the data and lead to more accurate and reliable forecasts. In the realm of anomaly detection, we combined LSTM autoencoder with OCSVM, utilizing the same principle of hyperparameter adjustment and rigorous data preprocessing, and achieved satisfactory results that outperformed several previous studies. The use of real-world datasets lends credibility to our results, in contrast to studies that use generated datasets. Although we applied our framework to address the task of lead time in the context of automobile distribution in Morocco, the principles and techniques we employed are transferable to other forecasting scenarios. Our framework can be adapted to various domains, such as finance, healthcare, and retail, to name a few examples. By utilizing our proposed models and methods, practitioners can enhance the accuracy and reliability of their forecasts across different contexts.

6.2 Limitations

Our study has several limitations that must be addressed to provide more comprehensive insights into the challenges of supply chain management. One major limitation is the reliance on a limited dataset that may not capture the full spectrum of supply chain scenarios. To gain a more comprehensive understanding of supply chain dynamics, it is crucial to expand the dataset to cover various industries and situations. This will enable us to identify patterns and anomalies across diverse contexts, improving the generalizability of our approach.

Moreover, our research focuses on a select group of deep learning models, which may overlook other promising artificial intelligence techniques that could yield better results. Future studies should explore other machine learning models and techniques in order to identify the most effective approaches for lead time forecasting and anomaly detection.

Furthermore, the deep learning models employed in this study may be susceptible to issues arising from imbalanced data or bias, affecting their overall performance. Imbalanced data can result in biased predictions and poor generalization, while biases introduced during data collection, preprocessing, or model architecture selection can impede the model's ability to effectively recognize certain anomalies or patterns, ultimately limiting the model's applicability. Therefore, it is essential to evaluate and compare different data preprocessing techniques and bias detection methods to minimize these issues. This can involve exploring different sampling techniques, data augmentation methods, or regularization strategies to address imbalanced data, as well as performing sensitivity analysis to detect and mitigate biases introduced at various stages of the model development process.

6.3 Conclusion and Future Works

In conclusion, this study addresses two critical challenges in supply chain management: lead time forecasting and anomaly detection. Our proposed approach employs a CNN-BiLSTM model and combines an LSTM autoencoder with OCSVM for anomaly detection, producing promising results that demonstrate the potential of our method in addressing these complex tasks effectively. The versatility of our approach makes it applicable across various domains that require anomaly detection and forecasting. For future research, we plan to enhance our forecasting model's performance by incorporating additional data sources. Additionally, we will explore integrating various deep learning models and techniques, such as attention mechanisms and graph neural networks, to further improve forecasting accuracy and anomaly detection capabilities. Attention mechanisms can help our model focus on critical features that may have a significant impact on lead time, while graph neural networks can effectively model the complex relationships between various entities in the supply chain, such as suppliers, manufacturers, distributors, and customers. By incorporating additional data sources and integrating various deep learning models and techniques, we can enhance our approach's forecasting and anomaly detection capabilities and facilitate more proactive decision-making and risk mitigation strategies.

REFERENCES

Aamer, A., Eka Yani, L., & Alan Priyatna, I. (2020). Data Analytics in the Supply Chain Management: Review of Machine Learning Applications in Demand Forecasting. *Operations and Supply Chain Management: An International Journal*, *14*(1), pp. 1-13.

Alzubaidi, L., Zhang, J., Humaidi, A. J., Al-Dujaili, A., Duan, Y., Al-Shamma, O., ... & Farhan, L. (2021). Review of Deep Learning: Concepts, CNN Architectures, Challenges, Applications, Future Directions. *Journal of big Data*, *8,* pp. 1-74.

Babai, M. Z., Dai, Y., Li, Q., Syntetos, A., & Wang, X. (2022). Forecasting of Lead-Time Demand Variance: Implications for Safety Stock Calculations. *European Journal of Operational Research*, *296*(3), *pp*. 46-861.

Balluff, S., Bendfeld, J., & Krauter, S. (2020). Meteorological Data Forecast using RNN. In *Deep Learning and Neural Networks : Concepts, Methodologies, Tools, and Applications* (pp. 905-920). IGI Global.

Boone, T., Ganeshan, R., Jain, A., & Sanders, N. R. (2019). Forecasting Sales in the Supply Chain : Consumer Analytics in the Big Data Era. *International Journal of Forecasting*, *35*(1), pp. 170-180.

Chandola, V., Banerjee, A., & Kumar, V. (2009). Anomaly Detection: A Aurvey. *ACM Computing Surveys (CSUR)*, *41*(3), pp. 1-58.

Chandriah, K. K., & Naraganahalli, R. V. (2021). RNN/LSTM with Modified Adam Optimizer in Deep Learning Approach for Automobile Spare Parts Demand Forecasting. *Multimedia Tools and Applications*, *80*(17), pp. 26145-26159.

Chen, Y., Fang, R., Liang, T., Sha, Z., Li, S., Yi, Y., ... & Song, H. (2021). Stock Price Forecast Based on CNN-BiLSTM-ECA Model. *Scientific Programming*, *2021*, pp. 1-20.

Chen, Y., Zhao, H., Zhou, R., Xu, P., Zhang, K., Dai, Y., ... & Gao, T. (2022). CNN-BiLSTM Short-Term Wind Power Forecasting Method Based on Feature Selection. *IEEE Journal of Radio Frequency Identification*, *6*, pp. 922-927.

Cooper, M. C., Lambert, D. M., & Pagh, J. D. (1997). Supply Chain Management : More Than A New Name for Logistics. *The International Journal of Logistics Management*, *8*(1), pp. 1-14.

Cortes, C., & Vapnik, V. (1995). Support-vector Networks. *Machine Learning*, *20*, pp. 273-297.

Ding, Z., & Fei, M. (2013). An Anomaly Detection Approach Based on Isolation Forest Algorithm for streaming data using Sliding Window. *IFAC Proceedings Volumes*, *46*(20), pp. 12-17.

Drissi, H., Lamzaouek, H., Amellal, I., & Mialed, K. (2022). Cash Flow Bullwhip Control Mechanisms in A Major Crisis Situation: A Case Study from the COVID-19 Crisis. *EuroMed Journal of Business*, (ahead-of-print).

Dudek, G. (2015). Short-term Load Forecasting using Random Forests. In *Intelligent Systems' 2014: Proceedings of the 7th IEEE International Conference Intelligent Systems IS'2014, September 24-26, 2014, Warsaw, Poland, Volume 2: Tools, Architectures, Systems, Applications* (pp. 821-828). Springer International Publishing.

Ellram, L. M., & Cooper, M. C. (1990). Supply Chain Management, Partnership, and the Shipper-Third Party Relationship. *The International Journal of Logistics Management*, *1*(2), pp. 1-10.

Fildes, R., Ma, S., & Kolassa, S. (2022). Retail Forecasting: Research and Practice. *International Journal of Forecasting*, *38*(4), pp. 1283-1318.

Glaser, A. E., Harrison, J. P., & Josephs, D. (2022). Anomaly Detection Methods to Improve Supply Chain Data Quality and Operations. *SMU Data Science Review*, *6*(1), pp. 3.

Greff, K., Srivastava, R. K., Koutník, J., Steunebrink, B. R., & Schmidhuber, J. (2016). LSTM : A Search Space odyssey. *IEEE Transactions on Neural Networks and Learning Systems*, *28*(10), pp. 2222-2232.

Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term Memory. Neural computation, 9(8), pp. 1735-1780. https://doi.org/10.1162/neco.1997.9.8.1735

Hu, W., Gao, J., Li, B., Wu, O., Du, J., & Maybank, S. (2018). Anomaly Detection using Local Kernel Density Estimation and Context-Based Regression. *IEEE Transactions on Knowledge and Data Engineering*, *32*(2), pp. 218-233.

Jaipuria, S., & Mahapatra, S. (2021). A Hybrid Forecasting Technique to Deal with Heteroskedastic Demand in a Supply Chain. *Operations and Supply Chain Management: An International Journal*, *14*(2), pp. 123-132.

Kerdprasop, N., Chansilp, K., Kerdprasop, K., & Chuaybamroong, P. (2019, June). Anomaly Detection with Machine Learning Technique to Support Smart Logistics. In *Computational Science and Its Applications–ICCSA 2019: 19th International Conference, Saint Petersburg, Russia, July 1–4, 2019, Proceedings, Part I* (pp. 461-472). Cham: Springer International Publishing.

Kim, T., Suh, S. C., Kim, H., Kim, J., & Kim, J. (2018, December). An Encoding Technique for CNN-Based Network Anomaly Detection. In *2018 IEEE International Conference on Big Data (Big Data)* (pp. 2960-2965). IEEE.

Kwon, D., Natarajan, K., Suh, S. C., Kim, H., & Kim, J. (2018, July). An Empirical Study on Network Anomaly Detection using Convolutional Neural Networks. In *2018 IEEE 38th International Conference on Distributed Computing Systems (ICDCS)* (pp. 1595-1598). IEEE.

Lambert, D., Stock, J. R., & Ellram, L. M. (1998). *Fundamentals of Logistics Management*. McGraw-Hill/Irwin.

Latecki, L. J., Lazarevic, A., & Pokrajac, D. (2007, July). Outlier Detection with Kernel Density Functions. In *MLDM* (Vol. 7, pp. 61-75).

Lesouple, J., Baudoin, C., Spigai, M., & Tourneret, J. Y. (2021). Generalized Isolation Forest for Anomaly Detection. *Pattern Recognition Letters*, *149*, pp. 109-119.

Lindemann, B., Jazdi, N., & Weyrich, M. (2020, August). Anomaly Detection and Prediction in Discrete Manufacturing Based on Cooperative LSTM Networks. In *2020 IEEE 16th International Conference on Automation Science and Engineering (CASE)* (pp. 1003-1010). IEEE.

Loureiro, A. L., Miguéis, V. L., & da Silva, L. F. (2018). Exploring the Use of Deep Neural Networks for Sales Forecasting in Fashion Retail. *Decision Support Systems*, *114*, pp. 81-93.

Lu, H., Ma, X., Huang, K., & Azimi, M. (2020). Carbon Trading Volume and Price Forecasting in China using Multiple Machine Learning Models. *Journal of Cleaner Production*, pp. *249*, 119386.

Lu, W., Li, J., Li, Y., Sun, A., & Wang, J. (2020). A CNN-LSTM-based Model to Forecast Stock Prices. *Complexity*, *2020*, pp. 1-10.

Maleki, S., Maleki, S., & Jennings, N. R. (2021). Unsupervised Anomaly Detection with LSTM Autoencoders using Statistical Data-Filtering. *Applied Soft Computing*, pp. *108*, 107443.

Manaswi, N. K. (2018). RNN and LSTM. *Deep Learning with Applications Using Python: Chatbots and Face, Object, and Speech Recognition With TensorFlow and Keras*, pp. 115-126.

Masci, J., Meier, U., Cireşan, D., & Schmidhuber, J. (2011). Stacked Convolutional Auto-Encoders for Hierarchical Feature Extraction. In *Artificial Neural Networks and Machine Learning–ICANN 2011: 21st International Conference on Artificial Neural Networks, Espoo, Finland, June 14-17, 2011, Proceedings, Part I 21* (pp. 52-59). Springer Berlin Heidelberg.

Meisheri, H., Sultana, N. N., Baranwal, M., Baniwal, V., Nath, S., Verma, S., ... & Khadilkar, H. (2022). Scalable Multi-Product Inventory Control with Lead Time Constraints using Reinforcement Learning. *Neural Computing and Applications*, *34*(3), pp. 1735-1757.

Mentzer, J. T., DeWitt, W., Keebler, J. S., Min, S., Nix, N. W., Smith, C. D., & Zacharia, Z. G. (2001). Defining Supply Chain Management. *Journal of Business logistics*, *22*(2), pp. 1-25.

Mhamdi, L., McLernon, D., El-Moussa, F., Zaidi, S. A. R., Ghogho, M., & Tang, T. (2020, October). A Deep Learning Approach Combining Autoencoder with One-Class SVM for DDoS Attack Detection in SDNs. In *2020 IEEE Eighth International Conference on Communications and Networking (ComNet)* (pp. 1-6). IEEE.

Miao, K., Hua, Q., & Shi, H. (2021). Short-term Load Forecasting based on CNN-BiLSTM with Bayesian Optimization and Attention Mechanism. In *Parallel and Distributed Computing, Applications and Technologies: 21st International Conference, PDCAT 2020, Shenzhen, China, December 28–30, 2020, Proceedings 21* (pp. 116-128). Springer International Publishing.

Nam, K., Hwangbo, S., & Yoo, C. (2020). A Deep Learning-Based Forecasting Model for Renewable Energy Scenarios to Guide Sustainable Energy Policy: A Case Study of Korea. *Renewable and Sustainable Energy Reviews*, *122*, 109725.Nanduri, A., & Sherry, L. (2016, April). Anomaly Detection in Aircraft Data using Recurrent Neural Networks (RNN). In *2016 Integrated Communications Navigation and Surveillance (ICNS)* (pp. 5C2-1). Ieee.

Nguyen, H. D., Tran, K. P., Thomassey, S., & Hamad, M. (2021). Forecasting and Anomaly Detection Approaches using LSTM and LSTM Autoencoder Techniques with the Applications in Supply Chain Management. *International Journal of Information Management*, pp. *57*, 102282.

Nourbakhsh, Z., & Habibi, N. (2022). Combining LSTM and CNN Methods and Fundamental Analysis for Stock Price Trend Prediction. Multimedia Tools and Applications, pp. 1-31.

Pan, S., Trentesaux, D., McFarlane, D., Montreuil, B., Ballot, E., & Huang, G. Q. (2021). Digital Interoperability in Logistics and Supply Chain Management: state-of-the-art and Research Avenues towards Physical Internet. *Computers in Industry*, pp. *128*, 103435.

Platt, J. C., Shawe-Taylor, J., Smola, A. J., & Williamson, R. C. (1999). Estimating the Support of a High-Dimensional Distribution. *Technical Report MSR-T R-99–87, Microsoft Research (MSR)*.

Provotar, O. I., Linder, Y. M., & Veres, M. M. (2019, December). Unsupervised Anomaly Detection in Time Series using LSTM-Based Autoencoders. In *2019 IEEE International Conference on Advanced Trends in Information Theory (ATIT)* (pp. 513-517). IEEE.

Rifai, S., Vincent, P., Muller, X., Glorot, X., & Bengio, Y. (2011, June). Contractive Auto-Encoders: Explicit Invariance during Feature Extraction. In *Proceedings of the 28th International Conference on International Conference on Machine Learning* (pp. 833-840).

Roy, S. S., Mittal, D., Basu, A., & Abraham, A. (2015). Stock Market Forecasting using LASSO Linear Regression Model. In *Afro-European Conference for Industrial Advancement: Proceedings of the First International Afro-European Conference for Industrial Advancement AECIA 2014* (pp. 371-381). Springer International Publishing.

Schölkopf, B., Burges, C. J., & Smola, A. J. (Eds.). (1999). *Advances in Kernel Methods: Support Vector Learning*. MIT press.

Siami-Namini, S., Tavakoli, N., & Namin, A. S. (2019, December). The Performance of LSTM and BiLSTM in Forecasting Time Series. In *2019 IEEE International Conference on Big Data (Big Data)* (pp. 3285-3292). IEEE.

Smagulova, K., & James, A. P. (2019). A Survey on LSTM Memristive Neural Network Architectures and Applications. *The European Physical Journal Special Topics*, vol. 228, no 10, pp. 2313-2324.

Sutskever, I., Vinyals, O., & Le, Q. V. (2014). Sequence to Sequence Learning with Neural Networks. *Advances in Neural Information Processing Systems*, *27*.

Thormann, M. (2015). Integration of Real-Time Demand Information and Spare Parts Distribution Planning for the Optimization of Spare Parts Supply in After-Sales Service Networks. *Operations and Supply Chain Management : An International Journal*, *8*(1), pp. 1-10.

Tran, K. P., Du Nguyen, H., & Thomassey, S. (2019). Anomaly Detection using Long Short Term Memory Networks and Its Applications in Supply Chain Management. *IFAC-PapersOnLine*, *52*(13), pp. 2408-2412.

Ullah, I., & Mahmoud, Q. H. (2022). Design and Development of RNN Anomaly Detection Model for IoT Networks. *IEEE Access*, *10*, pp. 62722-62750.

Vidal, A., & Kristjanpoller, W. (2020). Gold Volatility Prediction using a CNN-LSTM Approach. *Expert Systems with Applications*, pp. *157*, 113481.

Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y., Manzagol, P. A., & Bottou, L. (2010). Stacked Denoising Autoencoders: Learning Useful Representations in a Deep Network with a Local Denoising Criterion. *Journal of machine learning research*, *11*(12).

Zhang, L., Lin, J., & Karim, R. (2018). Adaptive Kernel Density-Based Anomaly Detection for Nonlinear Systems. *Knowledge-Based Systems*, *139*, pp. 50-63.

Asmae Amellal is currently pursuing her Ph.D. at the National School of Applied Sciences, Abdelmalek Essaadi University, in Tetouan, Morocco. She holds an engineering degree in electrical engineering from the Mohammedia School of Engineers (EMI) in Rabat, Morocco. She had over a decade of experience in the automotive distribution industry, as a development area manager in Morocco.

Issam Amellal is a Professor at the National School of Applied Sciences (ENSA) in Berrchid, Morocco. He obtained his Ph.D. in Logistics Engineering from the Faculty of Science and Technology of Settat in 2016. His main research interests are supply chain management and multi-criteria decision-making.

Hamid Seghiouer is a Professor and researcher at ENSATÉ, Tétouan, Morroco, holding a Ph.D. in Mathematics and Computer Science. His research focuses on information systems, decision support, and the application of mathematics to finance. In addition to his research, he is a dedicated educator, teaching courses and mentoring students at ENSATÉ.

Mohammed Rida ECH-CHARRAT received a Ph.D. degree in applied mathematics and computing, applied to reverse logistics from the National School of Applied Sciences of Tangier (ENSAT), Abdelmalek Essaadi University, Morocco. He is currently an Assistant Professor with the Abdelmalek Essaadi University attached to the National School of Applied Sciences of Tetouan (ENSATe). His research interests focus on optimization, supply chain management, data science, and artificial intelligence.