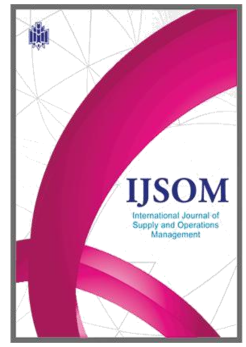
International Journal of Supply and Operations Management

IJSOM

2024, Volume 11, Issue 4, pp. 417-429
ISSN-Print: 2383-1359
ISSN-Online: 2383-2525
www.ijsom.com



A CNN–LSTM Hybrid Deep Learning Model for Detergent Products Demand Forecasting: A Case Study

Imen Ghazouani [a], Imen Masmoudi [a], Imen Mejri [b] and Safa Bhar Layeb [b*]

[a] *Industrial Engineering Department, National Engineering, University of Tunis El Manar, Tunis, Tunisia*
[b] *LR-OASIS, National Engineering School of Tunis University of Tunis El Manar, Tunis, Tunisia*

Abstract

An accurate forecast of current and future demand is an essential initial step for almost all the facets of supply chain optimization, including inventory strategy, production scheduling, distribution management, and marketing policies. Simply put, a more accurate demand prediction can lead to a more optimized supply chain process, allowing for better inventory control and higher customer satisfaction. Classical demand predictions are based principally on qualitative approaches relying on data from experts' opinions; quantitative forecasts based on historical data through statistical and artificial neural network models or a mix of qualitative and quantitative techniques that is also widely used and has shown good performances. Detergents and cleaning products demand is extremely volatile and has undergone substantial change, especially during the COVID-19 health crisis. In this paper, we present a hybrid Neural Network approach for accurate demand forecasts of the detergent manufacturing industry. It mainly consists of the combination of Long Short-Term Memory (LSTM) with Convolution Neural Network (CNN) based approaches. We performed a series of experiments on real data sets and assessed the performance of the proposed CNN–LSTM hybrid model. Numerical results showed that the combination of LSTM layers with complementary CNN layers provides more accurate results than other state-of-the-art forecasting models.

Keywords: Forecasting Demand; Detergent Products; Hybrid Neural Networks Based-Model; Convolution Neural Networks; Long Short-Term Memory.

I. Introduction

Prediction has historically been an integral part of decision-making and planning. The uncertainty facing the human environment is both an excitement and a challenge, as people and companies strive to reduce risks and optimize profits (e.g. Ouerghi et al., 2019). It is well-known that forecasting is a common statistical exercise in business and economic perspectives as it plays an essential role in numerous parts of an organization and serves to develop long-term investment and strategic planning policies (e.g. Mbarek et al., 2022). Many incorrectly believe that forecasting is not feasible in an uncertain environment. However, every situation may be subject to change, and an effective prediction approach detects data pattern change. Thus, forecasting hardly considers a static landscape. What is generally expected is that the pattern of how the situation is changing will be repeated in the future (Hyndman and Athanasopoulos, 2018).

*Corresponding author email address: safa.layeb@enit.utm.tn
DOI: 10.22034/ijsom.2024.109931.2752

Forecasting scenarios differ significantly in terms of the period or forecast horizon, parameters affecting current results, representations of data patterns, and numerous additional features. Modern companies require short-term, medium-term, and long-term predictions, according to their respective needs. According to Hyndman and Athanasopoulos (2018), short-term forecasts are required for personnel, production, and distribution planning. Demand predictions are also needed to support the scheduling activity. Medium-term predictions are required to identify the resource needs to buy raw materials, recruit employees or purchase new machines. Long-term predictions serve as a basis for strategic decision-making. This includes decisions that consider potential market changes, external environmental concerns, and available local resources.

The wide range of contexts requires a varied selection of predictive models and forecasting approaches to deal with real-world forecasting issues. A growing body of literature exists on forecasting theory and practice, such as the excellent survey initiated by DeGooijer and Hyndman (2006). This comprehensive review covers the spectrum of time series forecasting, from the most theoretical to the most widely applicable. The authors point out that the domain of time series forecasting has changed considerably since the 1980s. It has expanded with the enormous progress of computing efficiency, enhanced statistical techniques, and better-advanced methods for forecast processing and analysis. The increasing need to investigate large-scale data sets has fueled substantial interest in powerful computing techniques hence the focus of forecasters and decision-makers on advanced analytics and data science.

Classical statistical forecasting techniques such as the AutoRegressive Integrated Moving Average (ARIMA) model, have received much attention from forecasters to properly model a time series. More recently, with the progress of computer science, artificial neural networks and other approaches based on machine learning have become more and more popular for modeling time series (Brownlee, 2017). But much work remains to be accomplished, including accurate forecasts for numerous challenging problems that remain unresolved or have just emerged. On the other hand, Demand forecasting using deep learning models is still a relatively new field. New studies are presented and improved upon every year (e.g. Husna et al., 2021).

In the case of detergent manufacturing industry, the existing literature on the forecasting demand in this particular industry is very scant mainly due to the non-availability of dataset. To our best knowledge, the only work in this subject is the work of Mejri et al. (2021). The authors have introduced a Long Short-Term Memory (LSTM) based approach. The yielding results were compared to the results of the ARIMA model and have showed the good performance of the LSTM. In this work, we address a new hybrid approach to construct a deep learning model. The architecture of our model is founded on the idea that there should be a Convolutional Neural Network (CNN) layer, followed by layers for LSTM, and finally layers for extracting patterns from data. Thus, we propose to use deep learning to anticipate detergent demand in order to fulfill the specifications of its real implementation. The outcomes of the forecasts are contrasted with the state-of-the-art time series models and revealed the performance of the proposed CNN-LSTM model.

The remainder of this paper is organized as follows. Section II reviews the most related studies on demand forecasting. Section III presents the project context of the conducted study. Section IV compiles the new hybrid model. Section V reports the main numerical experiments and Section VI gives conclusions of the paper and provides avenues for future research.

II. Related Work

In this section, we start by providing an overview of time series forecasts and then describe the most widely applied forecasting methods and the main application contexts in the existing literature. We focus particularly on the techniques studied in this paper.

It is no worthy that time series forecasting is a strategy for projecting expected future patterns by looking at historical data and assuming that future events will reflect those of the past. Predicting refers to the practice of applying appropriate techniques based on previous data to forecast future results. When we are faced with forecasting problems that feature a time component, we can adopt time series forecasting which allows for performing data analysis procedures for robust and accurate planning (Armstrong, 2001).

1. Demand Forecasting Using Statistical Models

Statistical models are based on the generation of the rolling average for future demand. In the scenario of a mid-long-term forecasting for a family of products within a stable demand, these models demonstrated their viability and performance (Nguyen et al., 2020). The most widely used methods are: Moving Average (MA), Exponential Moving Average (EMA), Auto-Regressive Integrated Moving Average (ARIMA).

Moving Average (MA) is one of the most used statistical methods (Hansun, 2013). It's entirely dependent on mathematics (Hyndman, 2011). The MA process involves calculating several averages, each of which corresponds to a certain trend value for a specific time frame. The term "moving average" describes the average being calculated as the previous value is being replaced and the data from the following point is being displayed. This strategy is entirely self-sufficient (Brown et al., 2013).

Exponential Moving Average (EMA) is a similar technique to MA, but unlike this method, which provides equal weight to all data values, EMA gives higher weight to recent demand data (Hyndman and Athanasopoulos, 2018), which makes it a better indicator of a trend; thus, delivering accurate forecasts in a variety of time series in a short amount of time; which is a key advantage for industrial applications. For this type of method, the three main variants are: simple exponential smoothing, double exponential smoothing, and triple exponential smoothing.

Auto-Regressive Integrated Moving Average (ARIMA), is a technique that has gained a lot of attention for uni-variate time series demand forecasting (Weng et al., 2019). Unlike EMA which describes the trend detected in the data, the ARIMA model tackles autocorrelations in the data by using lagging indicators. Furthermore, Seasonal Autoregressive Integrated Moving Average (SARIMA) is an extension of ARIMA with the ability to support time series with a seasonal component since ARIMA only handles trends.

Despite their differences, all of these models share the advantage of being quicker and less expensive to implement than other forecasting techniques. However, that does result in less accuracy, which prompts the exploration of novel strategies to have accurate predictions, a pillar of the good management of the supply chain (Chopra and Meindl, 2016).

These time series forecasting approaches need stable data, i.e., data that exhibit predictable trends because otherwise, they risk failing to predict the market saturation of a product, as well as unexpected spikes in demand and seemingly irrational changes in customer preferences (Zhang, 1998).

2. Demand Forecasting Using Deep Learning Models

Artificial Neural Networks (ANN) are used to predict time series utilizing a variety of historical observations as input. The weights that determine the output result are updated throughout numerous runs to reduce the overall prediction error. Data patterns like trends and seasonality are not a constraint on ANNs, unlike exponential smoothing algorithms (Stoll, 2020). Even nonlinear data patterns can be represented, and these patterns are directly learned from the input. Interdependence is rarely linear in our complex world; in fact, nonlinearity can be seen in many real-world projections. It is possible to approximate any continuous linear and nonlinear function with any degree of accuracy (Zhang et al., 1998), (Hornik et al., 1989). In the exiting literature, examples of neural network topologies have included adaptive linear neurons, multi-layer perceptrons, and radial basis functions. However, the bulk of studies uses a multilayer perceptron structure to predict demand (Funahashi, 1989).

One of the most researched deep learning models for time-series forecasting is Long Short-Term Memory (LSTM). The special memory-like activity of LSTMs, a subclass of recurrent neural networks, enables them to learn patterns more effectively (Shi et al., 2019). However, the training data has a significant impact on their advantages. It can be challenging for these models to function without a lot of high-quality data. Deep neural networks are quite inefficient with data, which makes them a challenging choice for demand forecasting applications (Yu et al., 2011) and motivate attempts to hybride them (Lu et al., 2020).

III. Research Context

Our host company is a manufacturer that holds worldwide leading positions in both industrial and consumer businesses. They offer a wide range of detergent products, including space cleaners, laundry detergents, fabric softeners as well as adhesives and sealants. Not all of these products are of the same brand, which makes it difficult to manage the supply chain of all product lines.

A highly accurate system for demand forecasting must be in place in order to assess the value of an inventory policy for such a global company. The current paper tackles the demand forecasting for a specific category of this company's products, precisely the laundry detergents.

At first glance, historical sales data needed to be gathered for the targeted products in order to understand the first case and second predict the monthly detergent level demand 12 to 18 months into the future. Our analysis of the problem of interest revealed a total of 4 families with items of various sizes and brands, totaling 6 products. In order to respect the privacy of the company; the 4 families are called as **N**-HS REG, **N**-HS MS, **D**-LS Reg, and **D**-LS SDM. The first letter herein represents the product's brand and the second component serves as a specification for the product line. Historical 4-year sales from January 2018 to May 2021, including Covid-19 data, were our data set. Table I shows the data collected from January to May 2021.

Table 1. Historical sales for the targeted products (Year 2021)

Description	**Jan-21**	**Feb-21**	**Mar-21**	**Apr-21**	**May-21**
N-HS Reg 320g	21642	10832	16075	14425	19059
N-HS MS 320g	17647	8282	15705	10617	19272
D-LS Reg 4 Kg	2264	897	2511	1759	1051
D-LS Reg 5 Kg	2363	788	536	1253	1069
D-LS SDM 4 Kg	3509	643	2394	1428	927
D-LS SDM 5 Kg	2035	1795	744	1847	1129

IV. The Proposed CNN-LSTM Model

To forecast the product demand, a hybrid model that combines CNN and LSTM is built. The relevant aspects of the demand for detergents are extracted using the CNN model; the output herein is the input for the LSTM, which computes the predictions using the data on detergent usage from the past and present. The structure of the LSTM used in this model is Bi-LSTM; standing for bidirectional LSTM; that is an improved LSTM and it gives better accuracy of detergent demand prediction.

1. The CNN-LSTM Model Architecture

The architecture of the proposed CNN-LSTM-based deep learning framework is depicted in Figure 1.

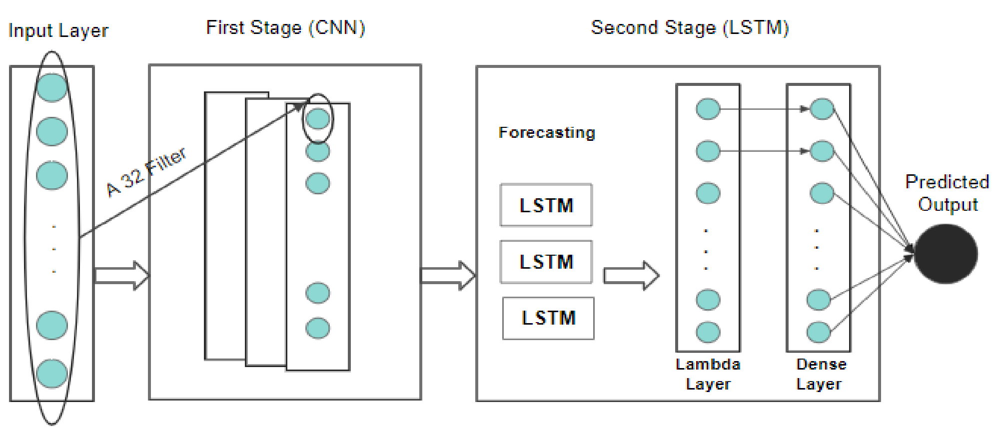


Figure 1. The structure of the proposed Hybrid CNN-LSTM model

More specifically, Figure 1 shows that the proposed model consists of an input layer, a one-dimensional convolutional layer, and three Bi-LSTM hidden layers with a recall function for the learning rate, which will be detailed later in the current paper, as well as a lambda layer and a dense layer.

- The **Con1D** layer: is a layer that creates a convolution kernel that is convolved with the layer input over one temporal dimension to produce a tensor of outputs.
- The **Bi-LSTM** layers: In bidirectional recurrent neural networks, each training sequence is presented both forward and backward to two separate recurrent nets that are coupled to the same output layer.
- The **Lambda** layer: is a layer that increases the LSTM's sensitivity to low values. Its parameter is determined in accordance with our findings.
- The **Dense** layer: is closely coupled to the layer above it, meaning that every neuron in the layer is also connected to every neuron in the layer above.

2. Data Preprocessing

As introduced in Section III, the used dataset is the monthly sales of six products from January 2018 to May 2021, totaling 41 values for each product. For sake of convenience, we renamed the products as P1-P6 to simplify the visuals. Table2 shows the correspondence of each product's name.

Table 2. Historical sales for the targeted products

Product	Designation
P1	N-HS Reg 320g
P2	N-HS MS 320g
P3	D-LS Reg 4 Kg
P4	D-LS Reg 5 Kg
P5	D-LS SDM 4 Kg
P6	D-LS SDM 5 Kg

2.1. Data Visualization

Let's begin by plotting the time series as displayed in Figure 2.

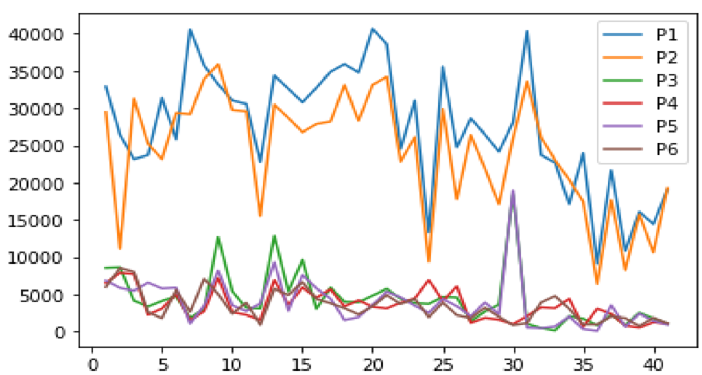


Figure 2. Sales of all the targeted products

Figure 2 shows that the product sales are on different scales. This points out the need to normalize the data to enable more accurate conclusions. The normalized data is shown in Figure 3.

After normalizing the data, we can observe in Figure 3 that P3 and P4 look to have an old pic around the 28th time step, which occurs to be around April 2020. This could be explained by the emergence of the coronavirus, in which case individuals would buy more detergent to wash their clothes in the belief that doing so would increase their protection, or it could be the result of a marketing campaign designed to boost sales through promotions so that more people buy. Overall, the data seems to be having a downward trend, and it can get influenced by external factors.

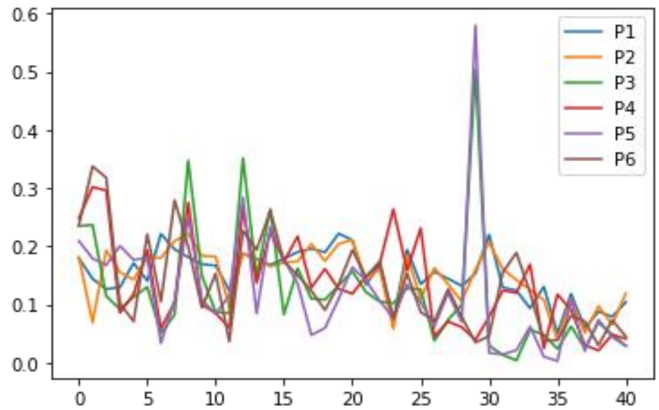


Figure 3. Normalized sales of all the targeted products

2.2. Correlation matrix

In order to gain insights into the correlation of sales between the different products, we had to visualize the correlation matrix using the heat map function from the Seaborne library.

Figure 4. The correlation matrix

Figure 4 displays the correlation matrix that reveals strong correlations between three couples of products: P1 and P2, P3 and P5, and P4 and P6. Going further to the essence of the products, the strong correlation between them became more clear. In fact:

• P1 and P2 are both "N-HS Deep Clean 320 Grams" the first is the regular and the second is MS.
• P3 and P5 are both "D-LS Powder 4 Kg New" the first is the regular and the second is MS.
• P4 and P6 are both "D-LS Powder 5 Kg New" the first is the regular and the second is MS.

We may infer from the product correlations that there is a significant influence between the pairs of products P1 and

P2, P3 and P5, and P4 and P6. In our model, we will include the couples that are correlated because the correlation will help the model better understand the pattern and provide more accurate forecasts.

2.3. Windowing the Dataset

We windowed the dataset to windows that contain in the inputs three values of each product's sales numbers and the outputs has one of each product's sales in the month after them.

```
import tensorflow as tf
def windowed_dataset(series, batch_size, n_past=24, n_future=24, shift=1):
    ds = tf.data.Dataset.from_tensor_slices(series)
    ds = ds.window(size=n_past + n_future, shift=shift, drop_remainder=True)
    ds = ds.flat_map(lambda w: w.batch(n_past + n_future))
    ds = ds.map(lambda w: (w[:n_past], w[n_past:]))
    return ds.batch(batch_size).prefetch(1)
```

Figure 5. Code for windowing the dataset

Figure 5 shows the developed code using Python. The first line, from tensor slices, is one of the simplest ways to create a dataset from a list in Python. The second line, ds.window splits the dataset into small parts of size= nFuture + nPast with a shift of 1-time step when selecting the values to create the windows. The Drop remainder = True means that all of the window will be of the same length, which means when it gets to the end of the dataset it will stop at the last element that gives us the window of the size that we specified. The structure is now divided into batches of the primary size and with the map function, the elements are split into features and labels.

3. Hyper parameter Settings and Optimization

The architecture of the proposed CNN-LSTM model is founded on the idea that there should be a CNN layer, followed by layers for LSTM, and finally layers for extracting patterns from data. The parameters as well as the total number of layers in the deep neural network model are summarized in Figure 6.

```
Model: "sequential_7"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv1d_7 (Conv1D)           (None, 3, 32)             224

 bidirectional_21 (Bidirecti  (None, 3, 64)            16640
 onal)

 bidirectional_22 (Bidirecti  (None, 3, 64)            24832
 onal)

 bidirectional_23 (Bidirecti  (None, 3, 32)            10368
 onal)

 dense_7 (Dense)             (None, 3, 6)              198

 lambda_1 (Lambda)           (None, 3, 6)              0

=================================================================
Total params: 52,262
Trainable params: 52,262
Non-trainable params: 0
```

Figure 6. Structure and parameters of the CNN-LSTM model

To go further in seeking for the best performance of the proposed model, we continued to look for the best parameters and fine-tune them (Reimers and Gurevych, 2017). As the input is limited, we first needed to decide on the first layer in the model and discovered that one CNN layer was sufficient. Three LSTM layers were added later to extract a strong correlation between the input and the output. The three layers are made bidirectional so they could forecast things more accurately.

After establishing the model's architecture, preliminary tests were performed and had showed that the best performance is found when the size of the data windows are fixed to three. Thus, the model is using data from the previous three months to produce projections for the following three months.

Then, a learning rate decay function was applied to be used as a callback in a learning rate scheduler that activates every ten epochs to divide by two the learning rate that the optimizer utilizes to ensure that our model is steadily convergent towards a stable good model. As the training process progressed, the learning rate changed as shown in Figure 7.

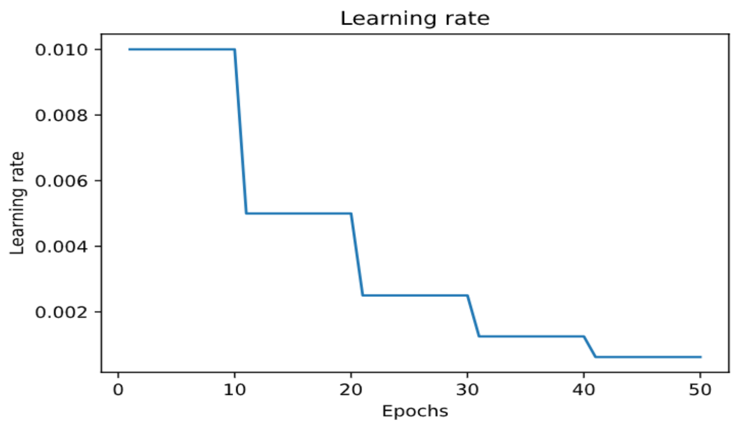


Figure 7. Evolution of the learning rate

In the initial run, we attempted to fit the model to the training data and let it run for 1,000 iterations to see how the training and evaluation losses changed. The plot that we derived is displayed in Figure 8.

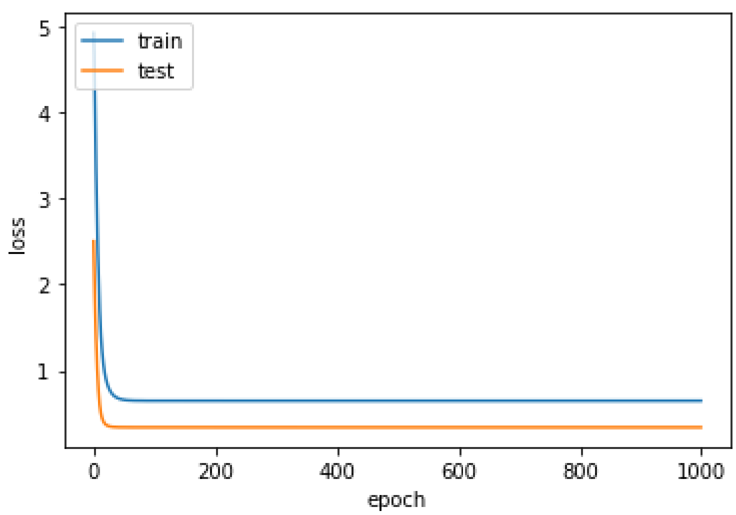


Figure 8. The number of epochs tuning

Figure 8 reveals that the model essentially stops evolving once it reaches one hundred epochs. Thus, the epochs number is set to 100 and the batch size to 10.

4. The Training

After setting up the model's architecture, the number of epochs, the learning rate scheduler, the batch size, and the windowed data that is split into training and validation datasets, we started the training process of our model.

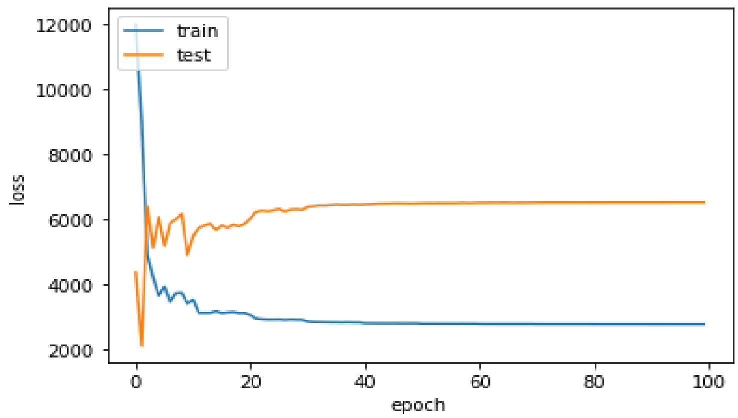


Figure 9. The history of loss for the training and testing set.

Figure 9 depicts how the loss changed as the number of epochs increased for both the training set and the testing set. As we can see, we began the training data's first epoch with a loss of more than 12000, which decreased over training to 3000 in the 20th epoch and reached a steady value of 2500 counting from the 21st epoch and onward.

V. Experimental Results and Analysis

To assess the proposed model performance, we used several metrics. Let's begin by denoting by y_i the actual/desired) value, $\hat{y_i}$ the estimated value and n the given data size, or predictions. The so-called Mean Squared Error (MSE) measures the average squared difference between the estimated values and the actual value. The MSE reads as

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y_i})^2. \quad (1)$$

The Mean Absolute Error (MAE) measures the average of error (difference between the estimated values and the actual value) in a set of predictions and is calculated as

$$\mathrm{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y_i}|. \quad (2)$$

The Root Mean Squared Error (RMSE) is the square root of the average of squared differences between prediction and actual values. Thus, the RMSE is equal to

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y_i})^2}. \quad (3)$$

The Mean Absolute Percentage Error (MAPE) measures the prediction accuracy of a certain forecasting technique and is calculated following the formula:

$$\mathrm{MAPE} = \frac{1}{n}\sum_{i=1}^{n}\frac{y_i - \hat{y_i}}{y_i}. \quad (4)$$

Considering Equations (1) -(4), the following values of the aforementioned metrics were found when applied to our model:

- ➔ **MSE** = 18694809.7388
- ➔ **MAE**= 2779.8286
- ➔ **RMSE**= 4323.7495
- ➔ **MAPE**= 45.207

At first glance, the proposed model appears to have been properly trained, giving confidence in the accuracy of the predictions. Therefore, Figure 10 shows the obtained forecast of our hybrid model for 12 months ahead starting from June 2021.

```
[[[30682.928  26903.994   4111.6846  3232.1406  3732.1284  3187.7305]
  [32600.205  28670.365   4224.6484  3436.6924  3770.1633  3345.9006]
  [30918.74   27240.533   3936.5261  3307.7507  3468.1206  3189.0579]]]
[[[30682.928  26903.994   4111.6846  3232.1406  3732.1284  3187.7305]
  [32600.205  28670.365   4224.6484  3436.6924  3770.1633  3345.9006]
  [30918.74   27240.533   3936.5261  3307.7507  3468.1206  3189.0579]]]
[[[30682.928  26903.994   4111.6846  3232.1406  3732.1284  3187.7305]
  [32600.205  28670.365   4224.6484  3436.6924  3770.1633  3345.9006]
  [30918.74   27240.533   3936.5261  3307.7507  3468.1206  3189.0579]]]
[[[30682.928  26903.994   4111.6846  3232.1406  3732.1284  3187.7305]
  [32600.205  28670.365   4224.6484  3436.6924  3770.1633  3345.9006]
  [30918.74   27240.533   3936.5261  3307.7507  3468.1206  3189.0579]]]
```

Figure 10. Demand forecasts using the CNN-LSTM model

Once the accuracy of the model highlighted, we will contrast its performance model with the state-of-the-art models, including ARIMA and LSTM (Mejri et al. 2021). To this end, we have focused on the average error MAPE and RMSE. Table 3 illustrates the results for each product and Table 4 shows the average of error on all the products.

Table 3. Comparison of the CNN-LSTM model with ARIMA/ LSTM for the month of February, March and April**.**

Metric	**Model**	**P1**	**P2**	**P3**	**P4**	**P5**	**P6**
Average MAPE	CNN-LSTM	15%	6%	10%	25%	12%	4%
	ARIMA	30%	47%	48%	26%	44%	39%
	LSTM	14%	22%	19%	27%	82%	47%
Average RMSE	CNN-LSTM	4441.38	1694.06	454.45	707.23	474.44	148.26
	ARIMA	8277.58	12724.33	1924.78	876.86	2136.35	1323.55
	LSTM	4609.21	7046.44	906.55	846.23	3897.82	1948.64

Table 4. Comparison of the CNN-LSTM model with ARIMA/ LSTM for the months of February, March, and April (average of error on all the products).

METRIC	MODEL	ALL PRODUCTS
AVERAGE MAPE	CNN-LSTM	**15%**
	ARIMA	34%
	LSTM	21%
AVERAGE RMSE	CNN-LSTM	2280.89
	ARIMA	7292.92
	LSTM	4167.29

Tables 3-4 shows that the hybrid model outperforms the ARIMA and LSTM models in terms of efficiency. For instance, for product P6, the CNN-LSTM model provides forecasts with an average MAPE of 4% compared to 39% and 47% for the ARIMA and LSTM, respectively. The overall average MAPE of the hybrid proposed model is about 15% while it is equal to 21% and 34% for the LSTM and ARIMA, respectively. Not surprisingly, ARIMA is lagging far behind the other models. Besides, the LSTM is providing better performance that ARIMA. However, the CNN-LSTM model consistently outperforms the two other models.

Similar to other neural networks, the hybrid CNN-LSTM model can learn linear and non-linear correlations as well as represent complicated interactions without imposing any assumptions on the distribution of the data. The effectiveness of the LSTMs was enhanced by the incorporation of the CNNs. Indeed, although the data was influenced by the hit of the Covid-19 virus, our model is more stable and adapts to the pattern of the data better than the other models. Therefore, the CNN-LSTM model appears to be more reliable and performant.

VI. Conclusion

This work investigates demand forecasting using a new hybrid CNN-LSTM model. The proposed deep learnin based model is applied for forecasting the demand of an international company of detergent industry. The architecture of the built model takes advantage of each of the CNN as well as the LSTM model. The numerical experiments revealed that the hybrid model outperforms the ARIMA and the LSTM models. Indeed, the proposed model gave accurate forecasting results and proved to be a reliable and efficient model to adapt in a real forecasting system implementation.

As an avenue for future research, the proposed CNN-LSTM model can be strengthened by altering the structure of the dataset and attempting to address the monthly demand per city instead of the overall monthly demand. Accordingly, the urban features (features that have an impact on sales) may be fed into the model such as characteristics of the city. Besides, customers' preferences for purchasing the product (their age and season of buying) could be taken into consideration. To go further in this investigation avenue, economic factors related to inflation and national financial crises, in addition to consumers' wages and salaries could be included in future work.

References

Abbasimehr, H., Shabani, M., & Yousefi, M. (2020). An optimized model using LSTM network for demand forecasting. *Computers & industrial engineering*, *143*, 106435.

Armstrong, J. S. (Ed.). (2001). *Principles of forecasting: a handbook for researchers and practitioners* (Vol. 30). Springer Science & Business Media.

Au, K. F., Choi, T. M., & Yu, Y. (2008). Fashion retail forecasting by evolutionary neural networks. *International Journal of Production Economics, 114*(2), 615-630.

Bandara, K., Shi, P., Bergmeir, C., Hewamalage, H., Tran, Q., & Seaman, B. (2019, December). Sales demand forecast in e-commerce using a long short-term memory neural network methodology. In International Conference on Neural Information Processing (pp. 462-474). Springer, Cham.

Box, G. E., Jenkins, G. M., Reinsel, G. C., & Ljung, G. M. (2015). *Time series analysis: forecasting and control*. John Wiley & Sons.

Brownlee, J. (2017). *How to Tune LSTM Hyperparameters with Keras for Time Series Forecasting*.

Chang, P. C., Wang, Y. W., & Tsai, C. Y. (2005). Evolving neural network for printed circuit board sales forecasting. *Expert Systems with Applications, 29*(1), 83-92.

Chang, P. C., Wang, Y. W., & Liu, C. H. (2007). The development of a weighted evolving fuzzy neural network for PCB sales forecasting. *Expert Systems with Applications, 32*(1), 86-96.

De Gooijer, J. G., & Hyndman, R. J. (2006). 25 years of time series forecasting. *International journal of forecasting, 22*(3), 443-473.

Fattah, J., Ezzine, L., Aman, Z., El Moussami, H., & Lachhab, A. (2018). Forecasting of demand using ARIMA model. *International Journal of Engineering Business Management, 10*, 1847979018808673.

Goodwin, P., Ord, J. K., Öller, L. E., Sniezek, J. A., & Leonard, M. (2002). *Principles of Forecasting: A Handbook for Researchers and Practitioners: J. Scott Armstrong (Ed.)*, (2001), Boston: Kluwer Academic Publishers.

Hansen, J. V., & Nelson, R. D. (1997). Neural networks and traditional time series methods: a synergistic combination in state economic forecasts. *IEEE transactions on Neural Networks, 8*(4), 863-873.

Husna, A., Amin, S. H., & Shah, B. (2021). Demand forecasting in supply chain management using different deep learning methods. In Demand forecasting and order planning in supply chains and humanitarian logistics (pp. 140-170). IGI Global.

Hyndman, R. J., & Athanasopoulos, G. (2018). *Forecasting: principles and practice*. OTexts.

Jiang, F., Yang, X., & Li, S. (2018). Comparison of forecasting India's energy demand using an MGM, ARIMA model, MGM- ARIMA model, and BP neural network model. *Sustainability, 10*(7), 2225.

J Rao, R. D., & Parikh, J. K. (1996). Forecast and analysis of demand for petroleum products in India. *Energy policy, 24*(6), 583-592.

Khashei, M., & Bijari, M. (2011). A novel hybridization of artificial neural networks and ARIMA models for time series forecasting. *Applied Soft Computing, 11*(2), 2664-2675.

Kim, M., Choi, W., Jeon, Y., & Liu, L. (2019). A hybrid neural network model for power demand forecasting. *Energies, 12*(5), 931.

Law, R., Li, G., Fong, D. K. C., & Han, X. (2019). Tourism demand forecasting: A deep learning approach. *Annals of Tourism Research, 75*, 410-423.

Luo, C. S., Zhou, L. Y., & Wei, Q. F. (2013). *Application of SARIMA model in cucumber price forecast*. In Applied Mechanics and Materials (Vol. 373, pp. 1686-1690). Trans Tech Publications Ltd.

Mbarek, H. B., Layeb, S. B., Aissaoui, N. O., Jaoua, A., & Hadj-Alouane, A. B (2022, September). Predicting Patient Arrival Rates in a Multi-Specialty Outpatient Department. Proceedings of the 3rd Asia Pacific International Conference on Industrial Engineering and Operations Management, Johor Bahru, Malaysia.

McKinney, W., Perktold, J., & Seabold, S. (2011). *Time series analysis in Python with statsmodels*. Jarrodmillman Com, 96-102.

Mejri, I., Bouzid, A., Bacha, S., & Layeb, S. B. (2021, September). Forecasting Demand Using ARIMA Model and LSTM Neural Network: a Case of Detergent Manufacturing Industry. In 2021 International Conference on Innovation and Intelligence for Informatics, Computing, and Technologies (3ICT) (pp. 346-353). IEEE.

Ouerghi, A., Hasni, M., Jaidi, Z., & Layeb, S. B. (2019). A new combined linear-artificial neural network-based model for accurate inflation forecasting in Tunisia. *International Journal of Decision Sciences, Risk and Management, 8*(4), 220-233.

Ramanathan, U., & Muyldermans, L. (2010). Identifying demand factors for promotional planning and forecasting: A case of a soft drink company in the UK. *International journal of production economics, 128*(2), 538-545.

Remus, W., & O'Connor, M. (2001). *Neural networks for time-series forecasting. In Principles of forecasting* (pp. 245-256). Springer, Boston, MA.

Rosenberg, R. D. (1982). Forecasting derived product demand in commercial construction. *Industrial Marketing Management*, *11*(1), 39-46.

Sen, P., Roy, M., & Pal, P. (2016). Application of ARIMA for forecasting energy consumption and GHG emission: A case study of an Indian pig iron manufacturing organization. *Energy*, *116*, 1031- 1038.

Weng, Y., Wang, X., Hua, J., Wang, H., Kang, M., & Wang, F. Y. (2019). Forecasting horticultural products price using ARIMA model and neural network based on a large-scale data set collected by web crawler. *IEEE Transactions on Computational Social Systems, 6*(3), 547-553.

Yu, Y., Choi, T. M., & Hui, C. L. (2011). An intelligent fast sales forecasting model for fashion products. *Expert Systems with Applications, 38*(6), 7373-7379.

Yildiz, H., DuHadway, S., Narasimhan, R., & Narayanan, S. (2016). Production planning using evolving demand forecasts in the automotive industry. *IEEE Transactions on Engineering Management*, *63*(3), 296-304.