Decision Making: Applications in Management and Engineering
Vol. 6, Issue 2, 2023, pp. 668-690.
ISSN: 2560-6018
eISSN: 2620-0104
DOI: https://doi.org/10.31181/dmame622023695

THE MEMORY CONCEPT BEHIND DEEP NEURAL NETWORK MODELS: AN APPLICATION IN TIME SERIES FORECASTING IN THE E-COMMERCE SECTOR

Filipe R. Ramos[1*], Maria Teresa Pereira[2,3], Marisa Oliveira [2,3] and Lihki Rubio[4]

[1] CEAUL-Centro de Estatística e Aplicações, Faculdade de Ciências, Universidade de Lisboa, 1749-016 Lisboa, Portugal
[2] ISEP-Instituto Superior de Engenharia do Porto, 4249-015 Porto, Portugal
[3] Associate Laboratory for Energy, Transports and Aerospace (LAETA-INEGI), Rua Dr. Rober-to Frias 400, 4200-465 Porto, Portugal
[4] Universidad del Norte, Km. 5 vía Puerto Colombia, Barranquilla, 081007 Atlántico, Colombia

Received: 20 February 2023;
Accepted: 13 July 2023;
Available online: 24 July 2023.

Original scientific paper

Abstract: *A good command of computational and statistical tools has proven advantageous when modelling and forecasting time series. According to recent literature, neural networks with long memory (e.g., Short-Term Long Memory) are a promising option in deep learning methods. However, only some works also consider the computational cost of these architectures compared to simpler architectures (e.g., Multilayer Perceptron). This work aims to provide insight into the memory performance of some Deep Neural Network architectures and their computational complexity. Another goal is to evaluate whether choosing more complex architectures with higher computational costs is justified. Error metrics are then used to assess the forecasting models' performance and computational cost. Two-time series related to e-commerce retail sales in the US were selected: (i) sales volume; (ii) e-commerce sales as a percentage of total sales. Although there are changes in data dynamics in both series, other existing characteristics lead to different conclusions. "Long memory" allows for significantly better forecasts in one-time series. In the other time series, this is not the case.*

Keywords: *E-Commerce, time series, deep neural network, forecasting, prediction error, computational cost.*

* Corresponding author.
E-mail addresses: frramos@fc.ul.pt (F. R. Ramos), mtp@isep.ipp.pt (M. T. Pereira), mjo@isep.ipp.pt (M. Oliveira), lihkir@uninorte.edu.co (L. Rubio)

1. Introduction

In an increasingly global economy, it is possible to see an increase in the competitiveness of organisations. Having scientific knowledge and accurate time series forecasting methods can lead to success. The articulation of statistical techniques and tools, combining mathematical and computational aspects, is manifested in explicit support for decision-making, especially in the forecasting (Ramos, 2021).

In Hang (2019), forecasting is recognised as a fundamental tool. The author points out that it is essential to create a competitive advantage where forecasting tools support proactive planning (e.g., production, business, financing, investments) or even contribute to more efficient management of resources. As in other sectors, e-commerce (electronic shopping) is no exception.

The e-commerce sector has been an object of interest for professionals and researchers. The Internet and the World Wide Web provide an additional channel for consumers to find, select and purchase products (Wang & Dai, 2004). Since the creation of AMAZON by Jeff Bezos in 1994, e-commerce has grown exponentially in the last decades due to the increase in technology, logistics efficiency, and globalisation.

The COVID-19 pandemic accelerated the growth of e-commerce due to the lockdown that countries implemented, favouring digital business activities that experienced a substantial increase (Modgil et al., 2022). This scenario pushed people towards online shopping, the first new electronic shopping experience for many.

Given this current paradigm, many researchers have studied several important issues related to e-commerce. Regarding the impact on customer behaviour and satisfaction after the COVID-19 pandemic, Higueras-Castillo et al. (2023) analyses the drivers and barriers of online channel usage intentions. The same authors also assess the implications for physical channels (based on modifying the Unified Theory of Acceptance and Use of Technology model, UTAUT-2) and identify the relevant segments of e-commerce consumers versus physical shoppers in the post-COVID-19 world. Wang and Dai (2004) propose a fuzzy constraint satisfaction approach for electronic shopping assistance based on satisfaction with each product. Martínez-López et al. (2022) investigated the role of return method and return fee on the buyer-seller relationship. Jiang and Benbasat (2014) examine the virtual product experience on the perception of diagnosticity and flow in e-commerce—related to pick-up point inventory Ren et al. (2022) proposes an integrated forecast-optimisation approach (Machine Learning – Quantile Regression, MLQR) to optimise the predictive shipping inventory of pick-up points, taking into account emergency shipping based on the historical transaction data of the online retailer. Compared to the original machine learning algorithms, MLQR can effectively increase the profits of online retailers. Atsalakis (2016) proposes a neuro-fuzzy technique for forecasting a new technology in shopping to overcome the drawbacks of neural networks for predicting electronic shopping. According to this author, neural networks have been successfully used for forecasting time series due to their significant characteristics in dealing with non-linear data with self-learning ability. However, neural networks suffer from the difficulty of dealing with qualitative information and the "black box" syndrome, which limits their application in practice. Experimental results also show that the neuro-fuzzy approach outperforms the other two conventional models (AR and ARMA).

From these examples, and paying particular attention to the forecasting methods used, some questions can be raised: (1) Which methods are most commonly used by professionals? (2) Which direction does the scientific literature point to? (3) Is forecasting accuracy the only criterion to consider when choosing a model?

Wilson and Spralls III (2018) assessed the perceptions of business professionals about the usefulness of forecasting techniques and the requirements for their use in real-world scenarios. They find that business, economics and finance experts prefer classical methods (such as auto-regressive, exponential smoothing or moving average models).

However, in time series analysis, some events can cause changes in the dynamics of the historical data, which increases the complexity of modelling and forecasting a time series (Chatfield, 2016). Moreover, a change in the behaviour of the series, translated mathematically by a disturbance in the model parameters, leads to an increase in forecasting error. Therefore, this structural instability can impact the forecasting performance of time series models. In particular, econometric forecasting models show poor performance in the presence of this type of structural breakpoints (Pesaran & Timmermann, 2004). Classical methods, in particular, have been shown to have these limitations.

Given the limitations of classical methods highlighted in the scientific literature and taking advantage of the computational advances made in recent years - thanks to the use of graphics processing units (GPUs) - scientific research has been directed towards the application of various artificial intelligence techniques, namely those based on machine learning (Ramos, 2021).

According to Cavalcante et al. (2016), methodologies in Artificial Intelligence have significantly contributed to advances in forecasting analysis. In this paradigm, Artificial Neural Network (ANN) methodologies, namely Deep Neural Networks (DNN), have been mentioned in the scientific literature as an up-and-coming option (Sezer et al., 2020; Tealab, 2020; Tkáč & Verner, 2016). This can be seen not only in the improvement of more primitive DNN structures (e.g., Multilayer Perceptron – MLP) but also in the search for new architectures (e.g., Recurrent Neural Networks – RNN, or even more robust as Long Short-Term Memory networks) with better forecasting quality.

Although there is a wide range of areas that benefit from DNN models, research highlights that success tends to focus on: (1) the decision-making process (e.g., manufacturing, supply chain, transportation, health); (2) financial difficulties and bankruptcies; (3) and stock price forecasting.

Regarding (1) manufacturing processes, an ANN model was developed for optimised ternary metal alloy electrodes to detect CH4 gas as a test case (Ghosal et al., 2021). About supply chain, Corsini et al. (2022) introduces a data-driven framework based on machine learning and metaheuristic optimization to dynamically select the most suitable replenishment strategy for a complex two-echelon (supplier-inventory-factory) supply chain (SC) problem with perishable product and stochastic lead times. In this study, the ability of the framework under the predictive and the optimization perspective is assessed (considering use of Artificial Neural Network and Particle Swarm Optimization) and a sensitivity analysis on the influence of replenishment parameters is presented as well. Considering transport problems, the Istanbul transit passenger demand number was used to build a real-world prediction model using MLP architectures, comparing it to other popular machine learning models (e.g., k-Nearest Neighbours, Linear Regression, Random Forest, Support Vector Machine) (Utku & Kaya, 2022). MLP has more successful than other machine learning algorithms in the majority of transportation lines, according to the experimental results. There are no references, in comparative terms, to the computational cost attributed to each model. Concerning health, more specifically COVID-19, many studies use DNN models to analyse and predict the spread of COVID-19 in cities and countries. For example, in the study by Utku (2023), an innovative hybrid deep learning model was developed and extensively compared with popular machine learning and deep learning models such as MLP, RNN and LSTM. The model developed showed promising results, but there need to be references to the computational cost compared with other models used. Another example of using different LSTM architectures is applied to the steelmaking process, where the clogging in the Submerged Entry Nozzle (SEN), responsible for controlling the steel flow in continuous casting, is one of the main problems faced by (Diniz et al., 2022). The authors point out that this can result in losses associated with the process yield and compromise the product quality. The LSTM architectures showed promising results, although there are no references to the implicit computational cost.

When discussing (2) and (3), for example, Costa et al. (2019), Lopes et al. (2021) and Ramos et al. (2018) report that RNN models (e.g., LSTM) can be promising for modelling and forecasting time series with structure breaks or with very irregular behaviour (such as time series related to financial markets). However, despite the excellent forecasting quality, Lopes et al. (2021) and Ramos et al. (2021) notes that these neural network architectures have a high computational cost. Due to the facts mentioned by these authors, further reflection is essential, combining the prediction power and computational cost of DNN models.

In short, following Hochreiter and Schmidhuber Field's (1997) work, several RNNs have been proposed in the literature based on methods for learning time dependencies. It is worth highlighting their applicability and the memory capacity of these networks (in particular LSTM, which can retain long-term past information) (Koutník et al., 2014). However, some LSTM architectures can perform better than others, often due to the network and data patterns (Jozefowicz et al., 2015). Greff et al. (2015) point out that some details in the data can have a more significant impact on the modelling and forecasting process than the structure of the neural network. Therefore, this work will first focus on understanding the concept of DNN memory and then on evaluating its effectiveness in time series analysis and forecasting.

On this basis, developing research focusing specifically (and comparatively) on DNN architectures is considered beneficial. Not only is it a current topic in the literature, but it is also in line with the growing interest in Artificial Intelligence domains, particularly the applicability of machine learning methodologies. Not only have researchers shown particular interest, but organisations have also sought to adapt to these methodologies. These facts justify the methodological options outlined in this research. Using data related to e-commerce (as they represent a recent change in the dynamics of historical data), the most straightforward DNN architectures (MLP networks) and more robust RNN architectures with "long memory", as in the case of LSTM architectures, are explored.

In summary, in line with state of the art outlined in the previous paragraphs and illustrated in Figure 1, the literature emphasises the potential of RNN architectures, which are considered more robust, but rarely refer (in comparative terms) to the implicit computational cost. Now, from a practical point of view, organisations (e.g., governments, companies) need timely results. Therefore, It is essential to understand whether using more robust (computationally intensive) networks is always justified. Furthermore, can guidelines be identified from the characteristics present in the data to allow for a priori recognition of the cases in which such use is justified? These aspects have yet to be much discussed in the literature, so this work aims to contribute to the scientific debate.

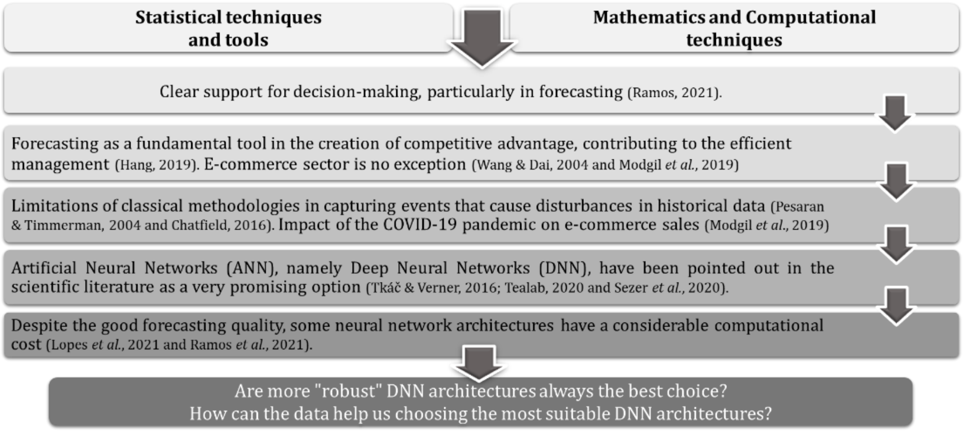
Statistical techniques and tools
Mathematics and Computational techniques
Clear support for decision-making, particularly in forecasting (Ramos, 2021).
Forecasting as a fundamental tool in the creation of competitive advantage, contributing to the efficient management (Hang, 2019). E-commerce sector is no exception (Wang & Dai, 2004 and Modgil *et al.*, 2019)
Limitations of classical methodologies in capturing events that cause disturbances in historical data (Pesaran & Timmerman, 2004 and Chatfield, 2016). Impact of the COVID-19 pandemic on e-commerce sales (Modgil *et al.*, 2019)
Artificial Neural Networks (ANN), namely Deep Neural Networks (DNN), have been pointed out in the scientific literature as a very promising option (Tkáč & Verner, 2016; Tealab, 2020 and Sezer *et al.*, 2020).
Despite the good forecasting quality, some neural network architectures have a considerable computational cost (Lopes *et al.*, 2021 and Ramos *et al.*, 2021).
Are more "robust" DNN architectures always the best choice?
How can the data help us choosing the most suitable DNN architectures?

Figure 1. Research question based on the state of the art.

In terms of the structure of this work, an overview of the literature, as well as the objectives, research questions and main contributions of this research, are provided in this extended introduction – Section 1. Section 2 reviews the DNN technique in the literature, particularly an explanation of implicit memory in neural networks. Next, section 3 presents the data to be used in this study and some considerations about the methodological procedures. Section 4 presents a descriptive and inferential data analysis, visualisations related to the predictions obtained by each model, and accuracy tables. Finally, Section 5 concludes the paper with a discussion of the results, conclusions, and references to limitations and future work.

2. Deep Neural Networks: Understanding the memory concept

The Multilayer Perceptron (MLP) network, which can be trained using the Backpropagation algorithm (Rumelhart et al., 1986), can be seen as one of the first steps towards DNN, as it presents multiple hidden layers of artificial neurons (Data Science Academy, 2019). In more complex architectures, such as RNNs, in addition to the learning that occurs in each training round, there is an additional learning input: the output of the neuron observed in the previous training round. The neuron can, therefore, capture this sequential learning. This type of architecture is based on an enhanced backpropagation algorithm, such as Backpropagation Through Time (BPTT) (Pineda, 1987). In addition, ANNs such as LSTMs, a subset of RNNs, can learn long-term dependencies and select which information to retain based on the data that allows the cost function to be minimised, which can be either more recent or older. Therefore, the excellent performance of this network in learning long-term dependencies is that: (i) it retains learning that occurred several time steps earlier (which is not the case for RNNs, which cannot retain long-term information); (ii) it forgets information that is not considered essential and therefore does not contribute to updating the network weights/biases. The differences in the learning process between MLP, RNN and LSTM architectures, which are all deep neural networks, occur essentially at the low level of the neuron, as shown in Figure 2.

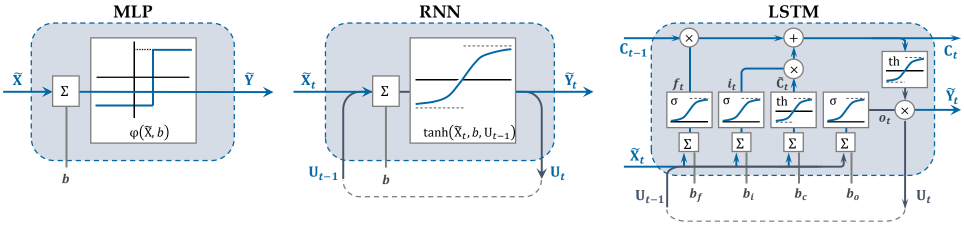

Figure 2. Comparison of hidden cells of MLP, RNN and LSTM.

MLP neurons are straightforward: each receives an input vector, $\tilde{X}$, and an external bias, b. These are then summed in a linear combination (Σ), and the output comes from this value passed through an activation function, φ, thus forwarding this output $\tilde{Y}$ to the subsequent neurons.

On the other hand, RNN neurons can capture information from previous training rounds. At each training round, t, each neuron is fed with an input vector, $\tilde{X}_t$, a bias, b, and a learning input that comes from the output of the previous training rounds, U_{t-1}. The activation function defines the output of the neuron. As mentioned above, and not only feeds the neurons in the next layer, $\tilde{Y}_t$, and provides feedback as input to the same neuron in the following training round, U_t. Looking at what happens within each neuron, it is apparent that it is possible to unroll RNN cells concerning time t $(t = 1, \dots, e$, where e represents the number of training rounds) – see Figure 3. This highlights that the feedback loop occurring in each neuron is not a feedback loop but the output of the same neuron in the previous training round (instant). The same reasoning applies to LSTM cells, the only difference being that LSTM has an output feedback loop and feeds back to the state cell.

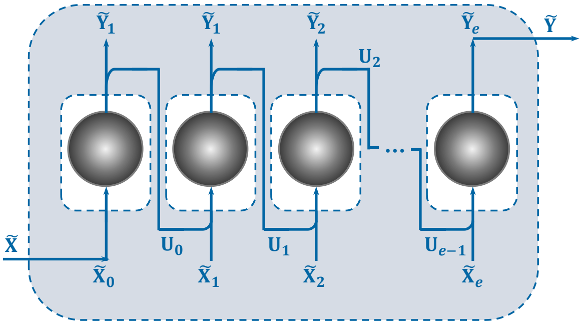

Figure 3. Comparison of hidden cells of MLP, RNN and LSTM.

In contrast to RNNs with short-term memory, LSTMs, with the help of the gates within the neurons, are prepared to capture both long-term and short-term memory and identify important information based on minimising the cost function. As shown and detailed in Figure 4, the state of the cell at each training round, t, acts as a memory kernel, as it can retain essential information throughout the processing sequence. The received input, C_{t-1}, corresponds to this kernel cell state in the previous training round. In contrast, C_t corresponds to the cell state value fed to the same neuron in the following training round. At each training round, t, neurons are provided by the input vector, $\tilde{X}_t$ the state of the same cell from the previous training round, U_{t-1}, and a bias, b, which is then processed within the neuron. The processing of these neurons distinguishes an LSTM cell from an RNN cell and gives the former its memory capabilities which can be exploited. LSTM cells use three types of gates: forget gate, input gate and output gate, each with some specificity.

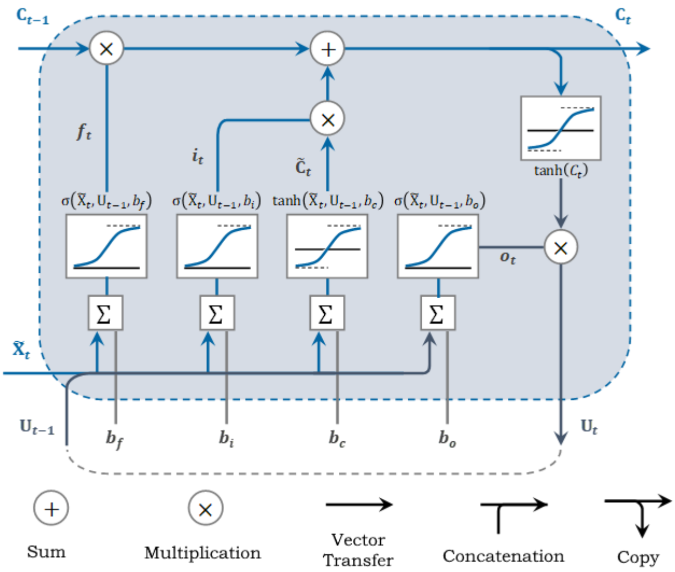

Figure 4. Hidden cells of LSTM architecture

1. In the first phase (forget gate), the information goes through a sigmoid function(σ) which results in the output of the Forget Gate, f_t, a scalar between 0 and 1, see Eq. 1.

$$f_t = \sigma(\tilde{X}_t,\ U_{t-1}, b_f) \tag{1}$$

This output will feed the cell state, multiplied by each value from the entrance of the input vector C_{t-1}. The state of the cell is then updated accordingly with

$$C_t = f_t \times C_{t-1} \tag{2}$$

2. On a second instance (input gate), the objective is to add new innovative information to the state cell, for which there are 3 steps. Initially, the information goes through a sigmoid function, which filters the values to be recalled and updated, as shown with the output i_t in Eq. 3.

$$i_t = \sigma(\tilde{X}_t,\ U_{t-1}, b_i) \tag{3}$$

Afterwards, a hyperbolic tangent function is responsible for constructing the new candidate state cells, $\tilde{C}_t$, with values between -1 and$+1$, which can be added to the state cell, as shown in Eq. 4.

$$\tilde{C}_t = tanh\ (\tilde{X}_t,\ U_{t-1}, b_c) \tag{4}$$

Finally, the values of such candidate vector $\tilde{C}_t$, and the processed values, i_t, are combined through multiplication to obtain a new vector of helpful information,

$i_t \times \tilde{C}_t$, which is added to the state cell, achieving the final cell state at each training round t, as shown in Eq. 5.

$$C_t = f_t \times C_{t-1} + i_t \times \tilde{C}_t \tag{5}$$

3. On a third instance (output gate) the information is filtered using a sigmoid function, which generates the output of this gate, o_t, as seen in Eq. 6.

$$o_t = \sigma(\tilde{X}_t,\ U_{t-1}, b_o) \tag{6}$$

The cell state vector referred to in the second instance, C_t, is transformed by applying a hyperbolic tangent function, generating a new vector with values between -1 and$+1$. Finally, the values from this new vector, $tanh(C_t)$, and the following filtered values, o_t, are multiplied until they are stored as input to feed the same neuron on the next training round, see Eq. 7.

$$U_t = o_t \times tanh(C_t) \tag{7}$$

This way, the LSTM network can distinguish between critical and non-important information and retain the former for a long time. This selective information storage allows the network to learn continuously throughout the training rounds.

3. Data and Methodology

3.1. Data

To carry out the empirical part of this study, two-time series related to e-commerce sales in the US were considered:[1]

- **Sales**: This time series refers to the sales volume (in millions of dollars) in e-commerce retail sales. The data presents a monthly frequency between January 2000 and November 2022 (275 observations);

- **Sales Ratio**: This time series refers to e-commerce retail sales as a percentage of total sales in the US. The data presents a quarterly frequency between January 2000 and July 2022 (91 observations).

Due to the COVID-19 pandemic, both time series show a disturbance in the historical data in 2020. Based on this similarity, to enrich the study (and to satisfy some of the proposed objectives), series with different volumes of data were used (which could affect the learning of the neural network). Furthermore, to assess the importance of memory in the neural network, the series show different dynamics after a perturbation in the historical data (in 2020). One series recovers the past dynamics (Sales series), while the other shows a different regime (sales ratio series). More details on the time series are discussed in section 4.1.

3.2. Methodological considerations

Regarding methodological procedures, Python language was used within the Jupyter Notebook environment, and all notebooks are available as open-source (Lopes

[1] Data were obtained from https://fred.stlouisfed.org/.

& Ramos, 2020). The important libraries used in the codebase were: numpy, pandas, statsmodels, matplotlib and tensorflow (with Keras integration).

In order to organise the development process, the code was separated into the following two notebooks: (1) *ExploratoryDataAnalysis.ipynb*, for exploratory data analysis, where first contact with the data is made in order to explore it and understand how it behaves (2) *DeepNeuralNetwork.ipynb*, which contains the code that implements the DNN models. All available notebooks have been developed from scratch based on the scientific literature. (e.g., Chollet, 2021; Ravichandiran, 2019).

Regarding DNN, the code allows the implementation of three architectures: MLP, RNN and LSTM. The approach taken to build the code was to: (i) pre-process the data before feeding it to the neural network; (ii) define the cross-validation methodology; and (iii) define the set of neural network hyper-parameters (e.g., number of layers, number of neurons per layer, number of training rounds, activation functions, optimisation algorithm, and others). A multi-grid explored several possible combinations, defining an accurate model. The sequential steps are shown in Figure 5.

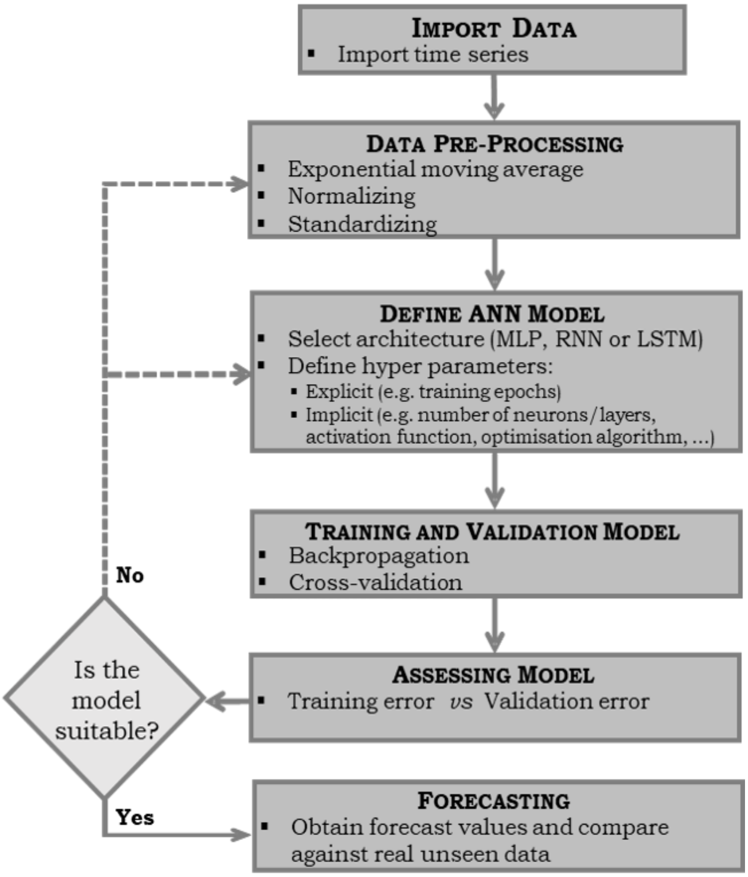
IMPORT DATA
▪ Import time series
DATA PRE-PROCESSING
▪ Exponential moving average
▪ Normalizing
▪ Standardizing
DEFINE ANN MODEL
▪ Select architecture (MLP, RNN or LSTM)
▪ Define hyper parameters:
▪ Explicit (e.g. training epochs)
▪ Implicit (e.g. number of neurons/layers, activation function, optimisation algorithm, ...)
TRAINING AND VALIDATION MODEL
▪ Backpropagation
▪ Cross-validation
ASSESSING MODEL
▪ Training error vs Validation error
Is the model suitable?
No
Yes
FORECASTING
▪ Obtain forecast values and compare against real unseen data

Figure 5. Methodology for computational implementation of DNN models.

A comparison of predicted values against actual price data that the model has not seen (test set) was required to assess the model. This analysis generates the forecast 'error'. The most common performance/error metrics are the following: Mean Absolute Error (MAE) and Mean Absolute Percentage Error (MAPE) (Willmott &

Matsuura, 2005). Considering the time series $\{y_t\}_{t\in T}$ and the past observations from period $1,\dots,t$, and being y_{t+h} an unknown value in the future $t+h$ and $\hat{y}_{t+h}$ its forecast, the prediction error corresponds to the difference between these two values, that is,

$$e_{t+h} = y_{t+h} - \hat{y}_{t+h} \tag{8}$$

where MAE and MAPE are defined, respectively, by

$$MAE = \frac{\sum_{i=1}^{s} |e_{t+i}|}{s} \tag{9}$$

$$MAPE = \frac{\sum_{i=1}^{s} \left|\frac{y_{t+i}-\hat{y}_{t+i}}{y_{t+i}}\right|}{s} \times 100 \tag{10}$$

where s corresponds to the number of observations in the forecasting samples (forecasting horizon).

4. Results: An application to the e-Commerce sector

In this research, the volume of sales (in millions of dollars) of e-commerce retail sales in the US (Sales time series) and e-commerce retail sales as a percentage of total sales in the US (Sales_Ratio time series) were considered – see Section 3.1.

4.1. Time series analysis

Regarding the Sales series, the data samples considered a monthly frequency between January 2000 and November 2022, i.e., a total of 275 observations (see Figure 6.)

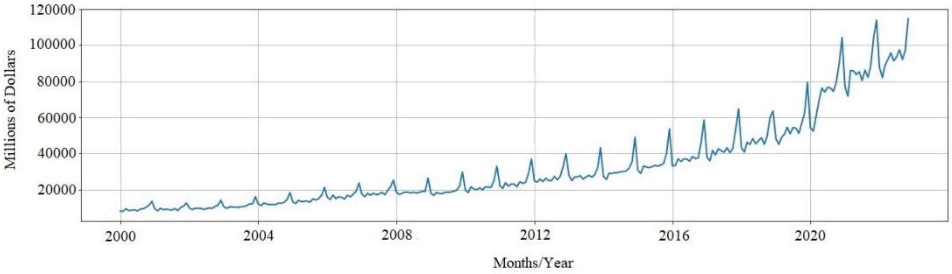

Figure 6. Sales time series (millions of dollars): graphical representation.

This graph shows an increasing linear trend, evident signs of seasonal behaviour (throughout the year), with a change in data dynamics in 2020. As a result of the COVID-19 pandemic, the social isolation policy adopted in March 2020 increased the volume of e-commerce sales in this period. The typical values, such as "peaks" in sales at the end of the year (November and December 2019), were also observed in 2020. In the last months of 2020 (November and December), the sales volume again increased significantly compared to the previous months. From then on, the data dynamics are very similar to those before 2020, albeit with a higher sales volume. This fact leads to a rightward skew in the distribution of historical sales volume data (see Figure 7). Additionally, the graphs of the decomposition of the time series (into additive and multiplicative components) and the correlogram are shown in Appendix A.

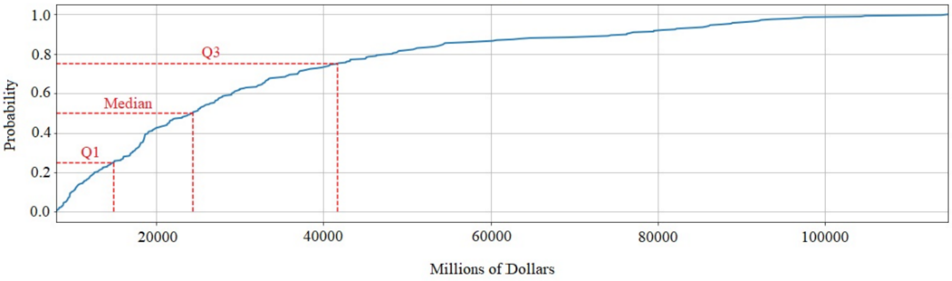

Figure 7. Sales time series (millions of dollars): graphical representation of cumulative distribution function.

For the quarterly Sales_Ratio series, the period was between January 2000 and July 2022, with 91 observations (see Figure 8).

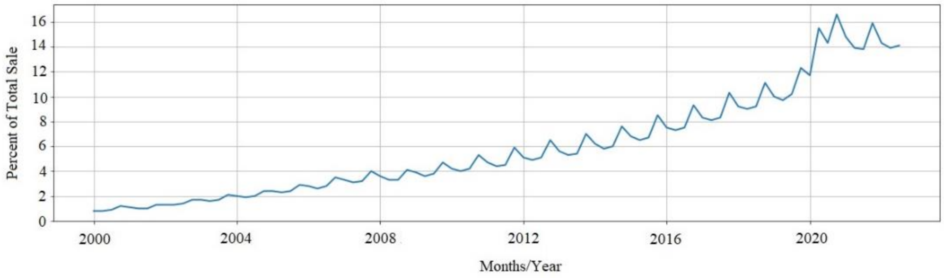

Figure 8. Sales_Ratio time series (percentage of total sales): graphical representation.

Although a globally increasing linear trend and seasonal behaviour can be observed in this case, there seems to be a regime change after 2020 (with the COVID-19 pandemic). The precise dynamics that the series followed until 2020 are lost, although a monotony pattern refers to the data history (namely between 2012 and 2020). These patterns can be better seen in the annual box plots (Figure 9), where a significant sample amplitude (corresponding to a variation of more than 5%) and a notable interquartile amplitude are observed in 2020 compared to other years. Outliers are still visible in the other years. They refer to the sales "peaks" at the end of each year (November or December), i.e. months in which there is an increase in sales volume.[2]

To analyse some features of the time series, Table 1 contains the statistic test and the $p - value$ for the following hypothesis tests: Normality tests (Jarque-Bera test and Skewness and Kurtosis tests), Stationarity/Existence of unit root (ADF test and KPSS test) and independence test (BDS test).[3]

[2] Additional information about the Sales_Ratio time series is presented in Appendix B (graphical representations of the decomposition of the time series and of the correlogram).

[3] For more details about all hypothesis tests, see Ramos (2021).

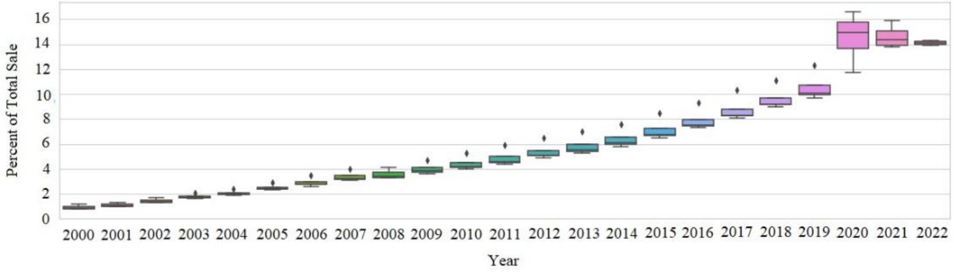

Figure 9. Sales_Ratio time series (percentage of total sales): graphical representation of annual box plots.

As expected, the normality, stationarity, and independence tests are rejected for any significance level for the two series under study. In fact, for both time series, there is statistical evidence to: (i) not reject the non-normality of the distribution of the data (with the rejection of the null hypothesis, which indicates normal behaviour in almost all tests performed); (ii) assume the non-stationarity (due to the null hypothesis not being rejected when doing ADF test, and due to the statistical value corresponding to KPSS being superior to the critical reference values); (iii) infer about the non-*iid*, since the null hypothesis of the data being *iid* has been rejected through BDS test.

Table 1. Sales and Sales_Ratio time series: normality, stationarity and independence tests.

		Normality Tests			**Unit Root / Stationary Tests**		**Independence Test**
		Kurtosis	*Skewness*	*Jarque-Bera*	ADF	KPSS	BDS (Dim.2–Dim.6)
Sales	*statistic*	2.9534	7.5036	105.4438	1.9890	1.4446	27.365 – 33.3505
	p-value	0.0031*	0.0000*	0.0000*	0.9987	---------	0.0000*
Sales Ratio	*statistic*	-0.0509	3.2667	11.9442	2.5186	0.7639	23.2494 – 32.8662
	p-value	0.9594	0.0011*	0.002*	0.9991	---------	0.0000*

* H_0 is rejected for significance levels of 1%, 5% and 10%

4.2. Modelling and Forecasting

In the process of modelling and forecasting time series, deep learning methods were considered. Specifically, the MLP and LSTM architectures were supposed to ensure the performance of time series forecasting. For these neural network models, some assumptions about the many combinations of different hyper-parameters involved had to be made. The analysis is carried out by considering: (A) inputs and data pre-processing; (B) DNN architectures and hyperparameters; (C) training, validation and evaluation of the models.

A. Inputs and pre-processing: Some work was done to verify that older/outdated information did not benefit the modelling process. In fact, not only did the computation time increase but so made the cross-validation errors. In terms of data pre-processing, by exponentially smoothing the historical data and then normalising it before feeding it to the ANN, it was found (from the cross-validation errors) that the prediction performance was improved.

B. DNN architectures and hyperparameters: The results of two different DNN architectures were analysed: MLP and LSTM. One of the main differences between these two architectures is undoubtedly the training time of the network (MLP takes minutes, while LSTM can take hours). There are many choices regarding the parameters and hyperparameters of the neural network. As mentioned in Section 3.2, starting from a baseline based on scientific literature, an exhaustive grid search was performed to explore different combinations of parameters and hyperparameters. See some details. The choice of the number of hidden layers and neurons per layer is a more subjective matter. From the studies carried out, the following were found to be good choices: (i) a number between three and five hidden layers; (ii) several neurons for the first and last hidden layers, close to the number of inputs and outputs, respectively; (iii) a higher number of neurons for the inner hidden layers. These choices proved appropriate, firstly because they did not interfere with learning the network and secondly because they avoided overfitting. In addition, the learning process showed that a reasonable (lower) number of neurons for the first hidden layer seemed to allow the network to capture the dynamics of the data. On the other hand, when many neurons were selected for the first hidden layer, the ANN caught any' noise', and the trained model did not give good results (more significant cross-validation errors). The remaining hyperparameters were chosen according to the ANN architecture used and following suggestions from the scientific literature. (e.g., Brownlee, 2018). The DNN models tend to overfit training data, so avoiding creating too complex models (too many neurons and/or layers) is essential. Some techniques explored to reduce overfitting were (i) regularisation (by adding an L1/L2 penalty to the cost function); (ii) dropout (by randomly dropping a set of neurons from a hidden layer at each training round); (iii) early stopping (by stopping training when the test error starts to increase).[4]

C. Training, validation and evaluation models: The ADAM optimiser was chosen to train the DNN, recognised as the most advanced optimiser (Kingma & Ba, 2015). The ADAM optimiser was selected to train the DNN and identified as the most advanced optimiser (Kingma & Ba, 2015). The stopping criteria were determined by playing with the number of rounds. Due to the dataset's characteristics and the ANN architecture (MLP or LSTM), it was found that between 150 and 200 training rounds seemed to give the best results, and this number of training rounds should be sufficient to stabilise training errors. As validation is an essential step in the model selection process, several attempts were made using Forward Chaining, K-Fold and Group K-Fold. Forward Chaining was found to be the most appropriate method in this study. The MAE forecasting performance metric was used to evaluate the model.

Therefore, for the Sales time series, a forecast is made for the following six months (from August 2022 to January 2023) using data up to July 2022 for modelling (training, validation). The aim is to assess the quality of the forecast in the short and medium term. Note that for the first four months (August to November), the forecasts were made within the time window of the historical data. In the case of December 2022 and January 2023, there is still no information on the actual values that occurred, so it is

[4] For more details, see Ramos (2021).

crucial to evaluate the performance of the models in light of the decline in expected sales volumes.

Figure 10 shows, for the sales time series, the performance of the two models to be evaluated (MLP model and LSTM model) and the future forecasts for the subsequent two observations (December 2022 and January 2023). The actual observations from August to November are shown (blue line) to compare the out-of-sample forecasts.

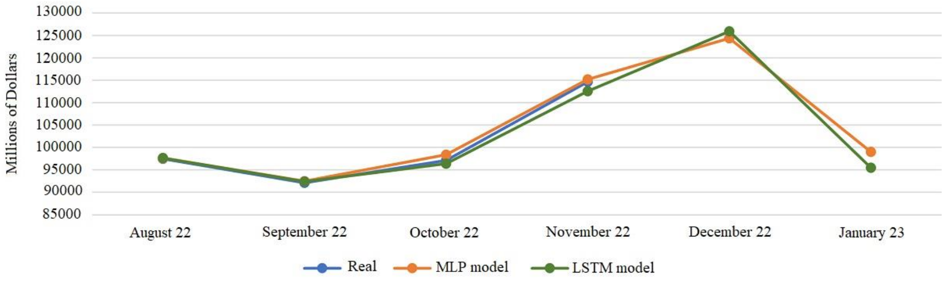

Figure 10. Sales time series (millions of dollars): forecasting performance of the DNN models and future projections.

While it is acceptable to mention that the LTMS model may have better accuracy, the two models have a similar forecasting quality (both in the short and medium term). Furthermore, both models can forecast the expected seasonal behaviour (increase in December and decrease in January).

The same methodology was applied to the Sales_Ratio time series. In this case, data up to July 2021 is used for modelling, and a forecast is made for the next six months (October 2021 to January 2023). For the first four cases (January to July 2022), forecasts were made within the time window of the historical data. There is still no information on the actual values that occurred for the following forecast values (October 2022 and January 2023), so it is important to evaluate the expected monotony.

In Figure 11, for each model ((A) MLP model and (B) LSTM model), it is possible to observe(i) data used to train the model (shaded area) to perform the in-sample prediction (orange line); (ii) data used to train the model to perform the out-of-sample prediction (blue line). These forecast values allow the performance of the model to be assessed.

From the analysis of both figures (Figure 10 and Figure 11), it may be concluded that both models generally capture the dynamics of the data and provide accurate forecasts. A more significant difference in the accuracy of the models is observed in the case of the Sales_Ratio time series. The LSTM model's predicted values follow the line of the actual values with greater precision than the MLP models. Furthermore, the expected seasonal behaviour (in December and January) seems to be more evident in the case of the LSTM model because of long-term memory.

4.3. Comparing Results

For a more detailed analysis, the MAPE values associated with the out-of-sample forecast have been calculated for four forecast observations (for which there is information about actual values). To present the MAPE values, it is vital to make one remark. The parameters of the neural network (weights and bias) benefited from a pseudo-random initialisation instead of using a fixed kernel (Glorot & Bengio, 2010).

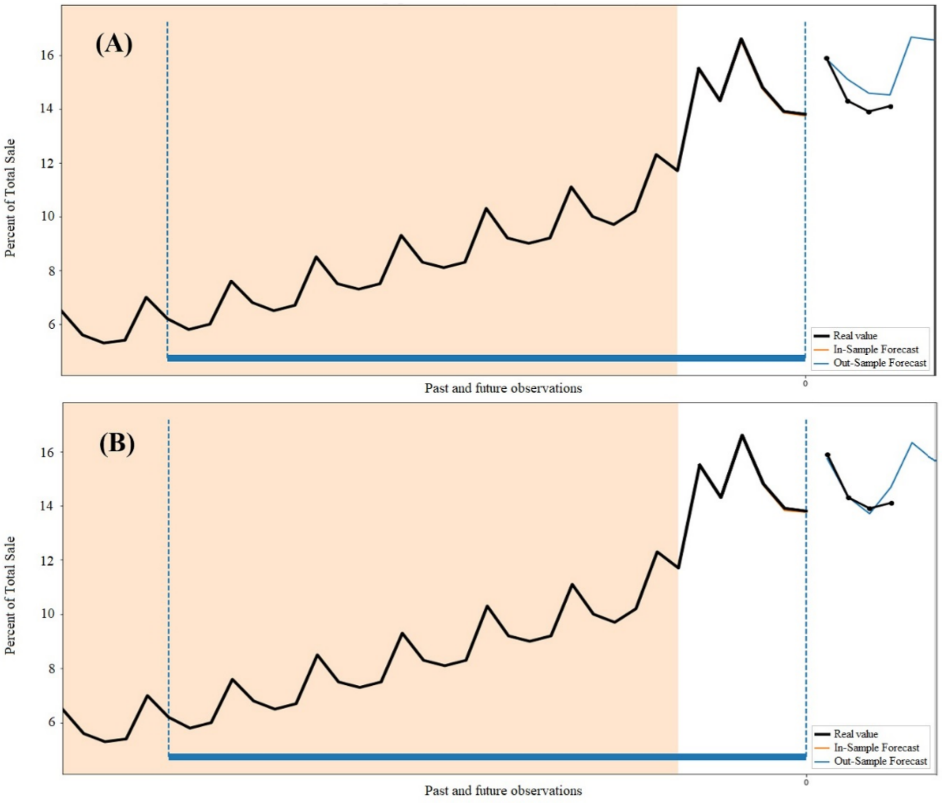

Figure 11. Fitting and forecasting of the Sales_Ratio time series: **(A)** MLP model; **(B)** LSTM model

For a more careful and fairer analysis and to avoid outlier results, the forecast was carried out in a loop (60 runs), and the 5% worst and best results were overlooked.

Table 2 shows the range of MAPE values (lower and upper bounds trimmed by 5%) for the MLP and LSTM models.

Table 2. Predictions errors of the MLP and LSTM models (MAPE)*

Model	Sales				Sales Ratio			
	Aug. 22	*Sept. 22*	*Oct. 22*	*Nov. 22*	*Oct. 21*	*Jan. 22*	*Apr. 22*	*July 22*
MPL	0.11% – 0.16%	0.21% – 0.32%	0.52% – 0.88%	0.97% – 1.38%	0.18% – 1.01%	1.89% – 2.47%	2.32% – 3.52%	2.80% – 3.87%
LSTM	0.10% – 0.16%	0.18% – 0.27%	0.40% – 0.58%	0.66% – 0.95%	0.18% – 0.99%	0.28% – 0.73%	0.31% – 0.92%	0.63% – 1.18%

[*] Minimum values - Maximum values (trimmed by 5%) obtained in a total of 60 runs

In general, focusing on the observed values, the following can be inferred: (i) the forecasting performance decreases with the increase of the time horizon for both DNN models (values ranging from 0.10% in the short term, reaching values above 3% in the medium term); (ii) the prediction errors generated by the LSTM forecast are generally more consistent (smaller error range) than those generated by the MLP forecast. In terms of forecasting quality: (i) both models have a good quality for the Sales time series (in the short term with similar forecasts); (ii) there is a more significant difference for the Sales_Ratio time series, with the LSTM model outperforming the MLP model; (iii) this difference is more marked as the forecasting horizon increases (for the LSTM models it reaches 1.18% in the worst case, whereas for the MLP models, it is above 2% in the best case). Given these differences (in both time series), the results must be examined in more detail.

Both time series show a change in the historical data dynamics, with a sharp and sudden increase in e-commerce due to the measures adopted due to the COVID-19 pandemic (e.g., social isolation and the physical shutdown of some commercial activities). However, in the case of the Sales time series, the dynamics recovered after the increase in 2020. With monthly data frequency, the observations available for model training allowed the MLP learning networks to understand this dynamic. As a result, the forecasting accuracy of the MLP model does not differ significantly from that of the LSTM model. In this case, these latter architectures' "long memory" capacity does not bring significant advantages. Past learning does not add any additional information to the model.

The same is not valid for the sales ratio time series. In this case, despite maintaining a particular seasonal behaviour, the trend line of the time series shows a change after 2022. With quarterly data frequency, the history of observations underlying this "new" behaviour is negligible. In this case, in addition to the change in the dynamics of the data, the reduced information available for training the neural network seems to be the reason for the poorer performance of the MLP model. This model tends to produce higher forecasts than those observed. On the other hand, the memories stored in the training and learning process, adapted to the new data dynamics, seem to help the LSTM networks to produce forecasts more closely aligned with the real data.

The above considerations confirm that the memory inherent in the LSTM architecture can, in some cases, play an important role in modelling and forecasting time series. It should be noted that these architectures are characterised by the use of memory resulting from past learning.

5. Discussions and Conclusions

This paper focuses on more advanced computational techniques for analysing and forecasting time series, given the limitations in the scientific literature concerning classical methods (e.g., the inability to handle more complex patterns and truly capture such dynamics).

A current and exciting topic in the scientific literature relates to e-commerce and the emerging paradigm shifts resulting from the "digital age" in which we live, which the COVID-19 pandemic has accelerated. In this context, the empirical study developed is based on two-time series relating to e-commerce retail sales in the US: (i) e-commerce sales volume; (ii) e-commerce sales as a percentage of total sales. Both time series show a change in dynamics due to the COVID-19 pandemic. Social isolation, travel restrictions, and the shutdown of some brick-and-mortar businesses, among other measures, led to a sharp and sudden increase in e-commerce.

It is interesting to analyse and forecast the volume of e-commerce sales and the weight of this volume to total sales. Here, forecasting methods play an essential role in supporting decision-making. In line with the literature, deep learning methods (in particular DNNs) were used to assess the 'learning' ability to extract relevant insights from data. "Long memory" neural networks (e.g., LSTM architectures) are proposed as the best option compared to other simpler neural networks (e.g., MLP architectures).

The results of this study are consistent with the literature. When analysing the forecasting ability, MLP models generally have a worse forecasting performance. The disadvantage of MLP networks is evident in cases where there is a change in the data dynamics, aggravated by the existence of few historical observations that enable better neural network training. In this case, confirming that the memory concept inherent to the LSTM architecture allows the network to learn the data better and improve the forecasts' quality is possible. The care taken in explaining the "memory concept" in Section 2 will enable us to understand why LSTM architectures stand out in this case. Not only referring to the predictive power of these architectures but understanding in detail where and also the real contribution as an asset of this investigation.

However, the choice of more robust neural network architectures, such as LSTM, is to have a reductionist view. This work brings to reflection a theme often neglected. In this research line, it is necessary to answer two fundamental questions, bringing a more profound reflection to the scientific literature: (1) Are more "robust" DNN architectures always the best choice?; (2) How can the data help us choose the most suitable DNN architectures?

It is often the case that only the quality of the forecast is considered, whilst the inherent computational cost is overlooked. Some (not much) literature highlights that long-memory neural networks, such as LSTM, have a high computational cost. This is because the choice of hyperparameters can be more complex, and learning and validation time is substantially higher (about 30% to 40% higher compared to MLP networks). Unless powerful machines are used, several hours of training may be required to benefit from the improved forecasting quality associated with LSTM models, which may not be feasible in a real-world context. Stakeholders (e.g., governments, investors, companies) sometimes require quick and timely responses.

In this study, although the time series may have suffered from disturbances in the historical data (which, according to the literature, can undermine the success of classical forecasting methods), the forecasting performance of MLP networks is similar to that of LSTM networks when the dynamics of the data are recovered. Thus, contrary to the prevailing idea of opting for more complex DNN models, simpler neural network architectures may be the right choice. Using our understanding of the concept of memory behind DNNs (and where this memory can be helpful in the training and validation process), combined with careful analysis of the data history, can be a sound practice. In this way, the quality of the forecast is preserved, and a significant reduction in computational cost is achieved. These considerations answer the above questions, which are often neglected in the literature. The predictive power is important, and the computational cost must be considered.

In short, the forecasting models should mind the needs of the real world.

From the above, and confident that this paper sheds light on the subject, some limitations are acknowledged in the present research. Namely: (i) further development of the analytical perspective – so that the results observed can have a more robust theoretical underpinning; (ii) increasing the diversity of the time series used – so that extrapolated conclusions are more robust and unbiased concerning the data used. Given these limitations, future work is suggested to explore the theory behind DNN models further and test the results on time series with different properties (without trend and/or seasonality). In addition, future work should explore improvements to the DNN models to reduce computational time significantly and, if possible, improve forecast accuracy. Research and implementation of hybrid models (e.g., Ramos et al., 2022; Rubio and Alba, 2022) have been highlighted in the literature as a promising solutions.

In general, their contributions are acknowledged, although some limitations are beyond this study's scope. It presents a contribution to the theory analysed (namely, the memory concept present in some DNN architectures) and, above all, to the computational aspect discussed. The robustness of the computational routines constructed (open source), the computational tests carried out, and their results' interpretation further contribute to the work on the subject matter. It is believed that this work can be used as a starting point for future work, where a compromise between the strengths of Artificial Intelligence and the human ability to understand what the data require is vital.

Author Contributions: All authors (F.R., M.P., M.O. and L.R.) contributed to the study's conception. Individual contributions: conceptualization, F.R..; literature review, F.R., M.T. and M.O.; methodology, F.R. and L.R..; computational implementation, F.R. and L.R.; research data, F.R., M.T. and M.O.; data analysis, F.R.; writing – original draft preparation, F.R.; writing – review and editing, F.R., M.P., M.O. and L.R.; final validation, F.R., M.P., M.O. and L.R.. All authors have read and agreed to the published version of the manuscript.

Funding*:* This work is partially financed by national funds through FCT – Fundação para a Ciência e a Tecnologia under the UIDB/00006/2020 project.

Data Availability Statement*:* The data used in this study can be obtained from https://fred.stlouisfed.org/.

Acknowledgements: This work is partially financed by national funds through FCT – Fundação para a Ciência e a Tecnologia under the UIDB/00006/2020 project.

Conflicts of Interest*:* The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper. The funders had no role in the study's design; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

Appendix A

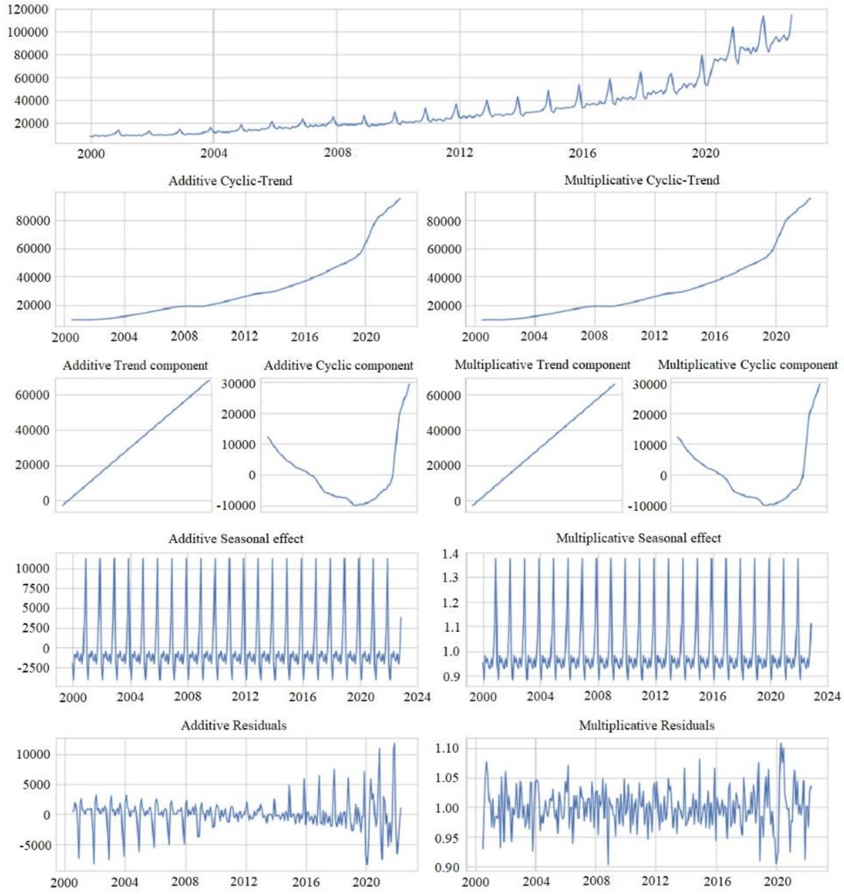

Figure A1. Sales time series: graphical representation of the decomposition

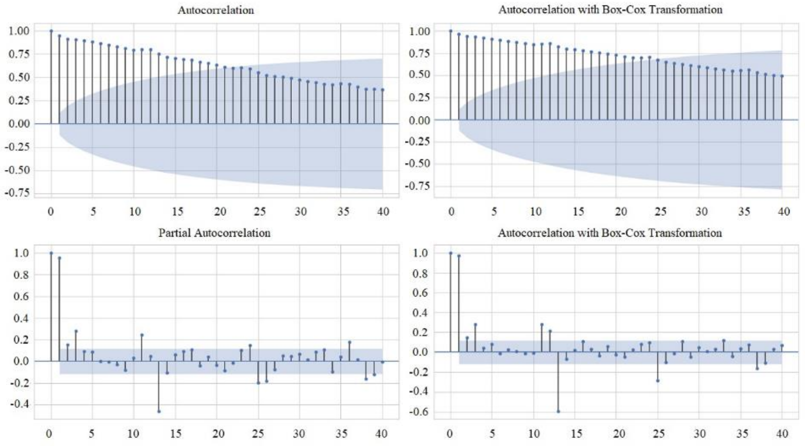

Figure A2. Sales time series: graphical representation of the correlogram

Appendix B

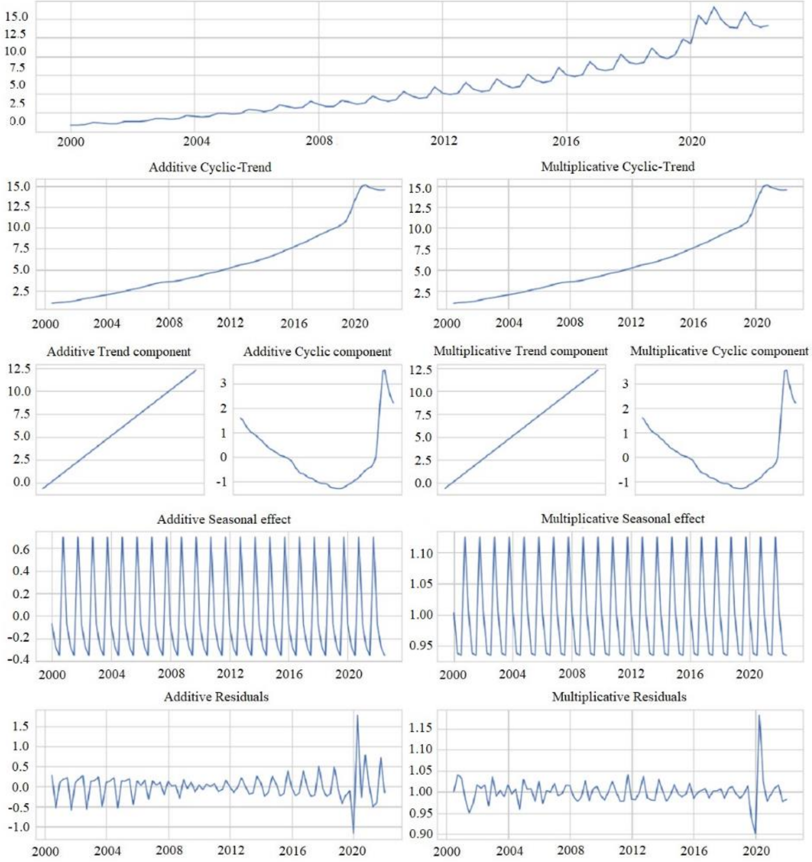

Figure B1. Sales_Ratio time series: graphical representation of the decomposition

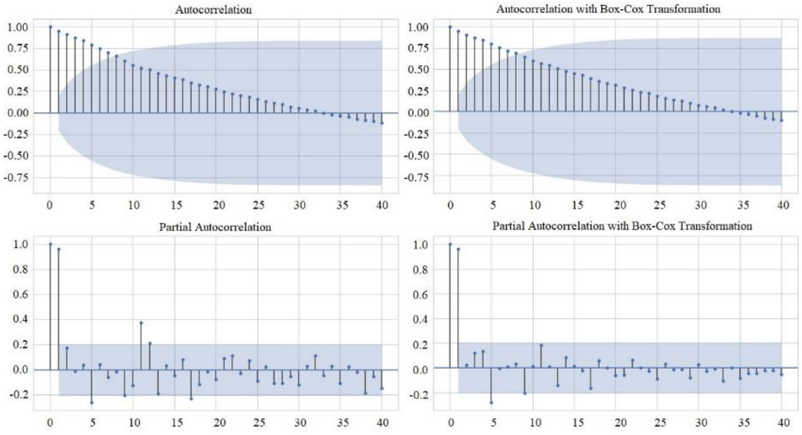

Figure B2. Sales_Ratio time series: graphical representation of the correlogram

References

Atsalakis, G. (2016). New Technology in Shopping: Forecasting Electronic Shopping With the Use of a Neuro-Fuzzy System. *Journal of Food Products Marketing*, *23*(5), 522–532. https://doi.org/10.1080/10454446.2014.1000445

Brownlee, J. (2018). *Deep learning for time series forecasting: predict the future with MLPs, CNNs and LSTMs in Python*. Machine Learning Mastery.

Cavalcante, R. C., Brasileiro, R. C., Souza, V. L. F., Nobrega, J. P., & Oliveira, A. L. I. (2016). Computational Intelligence and Financial Markets: A Survey and Future Directions. *Expert Systems with Applications*, *55*, 194–211. https://doi.org/10.1016/j.eswa.2016.02.006

Chatfield, C. (2016). *The Analysis of Time Series: an introduction* (6th ed.). Chapman and Hall/CRC.

Chollet, F. (2021). *Deep Learning with Python, Second Edition*. Manning Publications.

Corsini, R. R., Costa, A., Fichera, S., & Framinan, J. M. (2022). A new data-driven framework to select the optimal replenishment strategy in complex supply chains. *IFAC-PapersOnLine*, *55*(10), 1423–1428.

Costa, A., Ramos, F. R., Mendes, D., & Mendes, V. (2019). Forecasting financial time series using deep learning techniques. In *IO 2019 - XX Congresso da APDIO 2019*. Instituto Politécnico de Tomar - Tomar.

Data Science Academy. (2019). *Deep Learning Book*. Retrieved from http://deeplearningbook.com.br/

Diniz, A. P. M., Ciarelli, P. M., Salles, E. O. T., & Coco, K. F. (2022). Long Short-Term Memory Neural Networks for Clogging Detection in the Submerged Entry Nozzle. *Decision Making: Applications in Management and Engineering*, *5*(1), 154–168. https://doi.org/10.31181/dmame0313052022d

Ghosal, S., Dey, S., Chattopadhyay, P. P., Datta, S., & Bhattacharyya, P. (2021). Designing optimized ternary catalytic alloy electrode for efficiency improvement of semiconductor gas sensors using a machine learning approach. *Decision Making: Applications in Management and Engineering*, *4*(2), 126–139. https://doi.org/10.31181/dmame210402126g

Glorot, X., & Bengio, Y. (2010). Understanding the difficulty of training deep feedforward neural networks. In *JMLR W\&CP: Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics (AISTATS 2010)* (pp. 249–256). Sardinia: JMLR Workshop and Conference Proceedings.

Greff, K., Srivastava, R. K., Koutník, J., Steunebrink, B. R., & Schmidhuber, J. (2015). LSTM: A Search Space Odyssey. *IEEE Transactions on Neural Networks and Learning Systems*, *28*(10), 2222–2232. https://doi.org/10.1109/TNNLS.2016.2582924

Hang, N. T. (2019). Research on a number of applicable forecasting techniques in economic analysis, supporting enterprises to decide management. *World Scientific News*, *119*, 52–67.

Higueras-Castillo, E., Liébana-Cabanillas, F. J., & Villarejo-Ramos, Á. F. (2023). Intention to use e-commerce vs physical shopping. Difference between consumers in the post-COVID era. *Journal of Business Research*, *157*, 113622. https://doi.org/10.1016/j.jbusres.2022.113622

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

Jiang, Z., & Benbasat, I. (2014). Virtual Product Experience: Effects of Visual and Functional Control of Products on Perceived Diagnosticity and Flow in Electronic Shopping. *Journal of Management Information Systems*, *21*(3), 111–147. https://doi.org/10.1080/07421222.2004.11045817

Jozefowicz, R., Zaremba, W., & Sutskever, I. (2015). An Empirical Exploration of Recurrent Network Architectures. In *ICML - International Conference on Machine Learning*.

Kingma, D. P., & Ba, J. L. (2015). Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*. International Conference on Learning Representations, ICLR. https://doi.org/10.48550/arXiv.1412.6980

Koutník, J., Greff, K., Gomez, F., & Schmidhuber, J. (2014). A Clockwork RNN. *31st International Conference on Machine Learning, ICML 2014*, *5*, 3881–3889.

Lopes, D. R., & Ramos, F. R. (2020). Univariate Time Series Forecast. Retrieved from https://github.com/DidierRLopes/UnivariateTimeSeriesForecast

Lopes, D. R., Ramos, F. R., Costa, A., & Mendes, D. (2021). Forecasting models for time-series: a comparative study between classical methodologies and Deep Learning. In *SPE 2021 – XXV Congresso da Sociedade Portuguesa de Estatística*. Évora - Portugal.

Martínez-López, F. J., Feng, C., Li, Y., & Sansó Mata, M. (2022). Restoring the buyer–seller relationship through online return shipping: The role of return shipping method and return shipping fee. *Electronic Commerce Research and Applications*, *54*, 101170. https://doi.org/10.1016/j.elerap.2022.101170

Modgil, S., Dwivedi, Y. K., Rana, N. P., Gupta, S., & Kamble, S. (2022). Has Covid-19 accelerated opportunities for digital entrepreneurship? An Indian perspective. *Technological Forecasting and Social Change*, *175*, 121415. https://doi.org/10.1016/j.techfore.2021.121415

Pesaran, M. H., & Timmermann, A. (2004). How costly is it to ignore breaks when forecasting the direction of a time series? *International Journal of Forecasting*, *20*(3), 411–425. https://doi.org/10.1016/S0169-2070(03)00068-2

Pineda, F. (1987). Generalization of Back propagation to Recurrent and Higher Order Neural Networks. *Undefined*.

Ramos, F. R. (2021). *Data Science na Modelação e Previsão de Séries Económico-financeiras: das Metodologias Clássicas ao Deep Learning*. (PhD Thesis, Instituto Universitário de Lisboa - ISCTE Business School, Lisboa, Portugal). https://doi.org/10.13140/RG.2.2.14510.02887

Ramos, F. R., Costa, A., Mendes, D., & Mendes, V. (2018). Forecasting financial time series: a comparative study. In *JOCLAD 2018, XXIV Jornadas de Classificação e Análise de Dados*. Escola Naval – Alfeite. https://doi.org/10.13140/RG.2.2.11548.41606

Ramos, F. R., Lopes, D. R., Costa, A., & Mendes, D. (2021). Explorando o poder da memória das redes neuronais LSTM na modelação e previsão do PSI 20. In *SPE 2021 – XXV Congresso da Sociedade Portuguesa de Estatística*. Évora - Portugal.

Ramos, F. R., Lopes, D. R., & Pratas, T. E. (2022). Deep Neural Networks: A Hybrid

Approach Using Box&Jenkins Methodology. In *Innovations in Mechatronics Engineering II. icieng 2022. Lecture Notes in Mechanical Engineering* (pp. 51–62). Springer, Cham. https://doi.org/10.1007/978-3-031-09385-2_5

Ravichandiran, S. (2019). *Hands-On Deep Learning Algorithms with Python: Master deep learning algorithms with extensive math by implementing them using TensorFlow*. Packt Publishing Ltd.

Ren, X. X., Gong, Y., Rekik, Y., & Xu, X. (2022). Data-driven analysis on anticipatory shipping for pickup point inventory. *IFAC-PapersOnLine*, *55*(10), 714–718. https://doi.org/10.1016/j.ifacol.2022.09.491

Rubio, L., & Alba, K. (2022). Forecasting Selected Colombian Shares Using a Hybrid ARIMA-SVR Model. *Mathematics, Vol. 10, Page 2181*, *10*(13), 2181. https://doi.org/10.3390/math10132181

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature*, *323*(6088), 533–536. https://doi.org/10.1038/323533a0

Sezer, O. B., Gudelek, M. U., & Ozbayoglu, A. M. (2020). Financial time series forecasting with deep learning : A systematic literature review: 2005–2019. *Applied Soft Computing*, *90*, 106–181. https://doi.org/10.1016/j.asoc.2020.106181

Tealab, A. (2020). Time series forecasting using artificial neural networks methodologies: A systematic review. *Future Computing and Informatics Journal*, *3*(2). https://doi.org/10.1016/j.fcij.2018.10.003

Tkáč, M., & Verner, R. (2016). Artificial neural networks in business: Two decades of research. *Applied Soft Computing*, *38*, 788–804. https://doi.org/10.1016/j.asoc.2015.09.040

Utku, A. (2023). Deep learning based an efficient hybrid prediction model for Covid-19 cross-country spread among E7 and G7 countries. *Decision Making: Applications in Management and Engineering*, *6*(1), 502–534. https://doi.org/10.31181/dmame060129022023u

Utku, A., & Kaya, S. K. (2022). Multi-Layer Perceptron Based Transfer Passenger Flow Prediction in Istanbul Transportation System. *Decision Making: Applications in Management and Engineering*, *5*(1), 208–224. https://doi.org/10.31181/dmame0315052022u

Wang, J., & Dai, C. H. (2004). A fuzzy constraint satisfaction approach for electronic shopping assistance. *Expert Systems with Applications*, *27*(4), 593–607. https://doi.org/10.1016/j.eswa.2004.06.004

Willmott, C., & Matsuura, K. (2005). Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance. *Climate Research*, *30*(1), 79–82. https://doi.org/10.3354/cr030079

Wilson, J. H., & Spralls III, S. A. (2018). What do business professionals say about forecasting in the marketing curriculum? *International Journal of Business, Marketing, & Decision Science*, *11*(1), 1–20.