Middle East Technical University Informatics Institute

RESEARCH PROPOSAL

TIME SERIES FORECASTING WITH LONG SHORT TERM MEMORY (LSTM) ARCHITECTURE IN DEFENSE INDUSTRY COMPANY

Advisor name: İbrahim Soner Yıldırım

(METU)

Student name: Rüstem Ozan Özdemir

June 2021

Orta Doğu Teknik Üniversitesi Enformatik Enstitüsü

ARAŞTIRMA ÖNERİSİ

SAVUNMA SANAYİ FİRMASINDA UZUN KISA VADELİ BELLEK MİMARİSİ İLE ZAMAN SERİSİ TAHMİNLEME

Danışman adı: İbrahim Soner Yıldırım

(ODTÜ)

Öğrenci adı: Rüstem Ozan Özdemir

Haziran 2021

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF SYMBOLS / ABBREVIATIONS

ANFIS	Adaptive Neuro-Fuzzy Inference System
ANN	Artificial Neural Network
AR	Autoregressive
ARCH	Autoregressive Conditional Heteroscedastic
ARIMA	Autoregressive Integrated Moving Average
ARMA	Autoregressive Moving Average
ARIMAX	Autoregressive Integrated Moving Average with Explanatory Variable
BiLSTM	Bidirectional Long Short Term Memory
CNN	Convolutional Neural Network
DNN	Deep Neural Network
ERP	Enterprise Resource Planning
GARCH	Generic Autoregressive Conditional Heteroscedastic
KNN	K-Nearest Neighbors
LSSVR	Least Square Support Vector Regression
LSTM	Long Short Term Memory
MA	Moving Average
MSLE	Mean Squared Logarithmic Error
RBF	Radial Basis Function
RNN	Recurrent Neural Network
SARIMA	Seasonal Autoregressive Integrated Moving Average
SVM	Support Vector Machine
SVR	Support Vector Regression

1. INTRODUCTION

Forecasting future demand based on historical time series data related to demand is called time series forecasting. The purpose of Time Series Prediction is to predict the behavior of complex systems by examining at past patterns of the same phenomenon (Kochak & Sharma, 2015). Demand forecasting is a branch of time series forecasting that focuses on better understanding customer demands. Decision makers in all industries involving sales operations are willing to obtain more reliable demand forecasting since accurate demand forecasting reduces uncertainties in decision making. Especially in the manufacturing sector, successful demand forecasting provides structured information to business management mechanisms for the company such as inventory management, production scheduling and capacity management. For instance, demand forecasting reduces the risks regarding overstocking and understocking regarding inventory management. Overstocking occurs when forecast is higher than actual demand and it leads to increase in storage, labor, insurance costs and decrease in quality. Moreover, understocking occurs when forecast is lower than actual demand and it causes loss of sales and reducing customer satisfaction and business trustworthiness.

According to Shen et al. (2020), Autoregressive model , Autoregressive Integrated Moving Average model (ARIMA) , Support Vector Machine (SVM) , and neural networks-based approaches have been proposed as time series forecasting methods. In addition to single approaches, several hybrid approaches that is combination of single approaches have also been proposed . On the other hand, since the time series data related to manufacturing, finance and energy are usually non-linear and non-stationary, and these approaches cannot effectively extract enough information so as to achieve accurate time series forecasting results. Deep learning is a novel approach that can obtain a multi-levels representation of the original input by combining non-linear operations. Each layer of the deep neural network transforms one level of representation into a more abstract level. By the help of layer structure, deep learning can learn various type of features from input data. Moreover, one of its advantages is that it processes data without providing any assumptions, unlike traditional methods.

In this part, the sector of the data to be used will be briefly mentioned and the model to be used will be correlated with the data. The data to be used in the demand forecasting study will belong to one of the product family for the defense industry company. In defense industry environment, end products are produced based on contracts between customers and firms, and these contracts dictate strict rules especially on supplier base and due dates. Customers' late notification of their demands and their requests for finished products in a short time change the production and project prioritization and make resource management difficult. Due to the nature of the defense industry, it is expected that the demands will be aggressive and irregular, but the change in the previous program causes other important projects to be disrupted. In this study, in order to manage the demands correctly, estimating the demand correctly before the actual demand of the customer and making the appropriate production program and project prioritization will allow more effective management of the relevant resources. The irregularity of the demands, that is, the demand for high quantities in a short time or the absence of demand for a long time may occur.

The conditions of the sector in which the data is located, and the structure of the data bring to mind the solutions of the nonlinear time series data set. As mentioned above, it is thought that deep learning methods can produce more accurate predictions in revealing hidden information in nonlinear structures compared to traditional methods and therefore in time series estimation.

It was decided that deep learning methods are more suitable according to the data to be used by comparing traditional methods and deep learning methods. Now, it is necessary to determine which deep learning method will be used in the study. In this study, we propose a demand forecasting method based on multi-layer Long Short Term Memory (LSTM) networks since LSTM has capability to capture nonlinear patterns in time series data, while considering the inherent characteristics of non-stationary time series data. Although Artificial Neural Network (ANN) developed in the early days are successful in capturing nonlinear relationships, the associations between neurons in Recurrent Neural Network (RNN), branch of ANN, build up a cycle which permits signs to move in various ways. RNN provide short-term memory by storing the activations from each time step that provides technique for processing sequence data referring to time series data. On the other hand, RNN have disadvantages that are the vanishing gradient problem and exploding gradient problem, which makes model hard to train. The prevalent solution to overcome these problems is to use gated architectures such as LSTM which can perform long term memory information. Although different types of ANNs can capture nonlinear patterns of time series data, research have indicated that the ANNs with shallow architectures are unable to accurately model time series with a high degree of nonlinearity, longer range and heterogeneous characteristics (Sagheer & Kotb, 2019).

The possible contribution of this study is that we propose the demand forecasting framework, which has not been performed before in Turkey's leading defense industry company, that helps the company to anticipate the fluctuated demands before they are notified by the customer regarding meeting customer demands at the desired time. The second possible contribution is that this study will involve performance comparison of LSTM and other methods that are traditional statistical methods and other deep learning methods.

Our research question is that “Is LSTM the best of the models used in forecasting?”.

In Section 2, Literature Review, we analyzed the related work.

2. LITERATURE REVIEW

When the time series forecasting methods were surveyed at the literature, it has been observed that there are basically two main categories that are traditional statistical methods and deep learning methods (Khashei & Bijari, 2011 , as cited in Abbasimehr et al., 2020). In accordance with the chronological order, traditional methods will be examined before contemporary methods.

The examination of traditional methods will be examined in three main categories. These are linear models, linear model handicaps and non-linear models. Nonlinear models and their handicaps will be examined together.

Traditional time series data analysis techniques frequently use linear regressions to fit models and then a moving average to anticipate outcomes. The “Auto-Regressive Integrated Moving Average”, known as ARIMA, is the de facto standard for such techniques. This linear regression-based technique has evolved through time, and as a result, various versions of this model, such as SARIMA (Seasonal ARIMA), and ARIMAX (ARIMA with Explanatory Variable), have been developed. These models perform pretty well for short-term projections (the next lag), but their long-term projections suffer greatly. (Namini et al., 2019). According to Shen & Wynter (2012), parametric methods rely on statistical approaches such as auto-regressive moving average models, linear and nonlinear regressions to assume a given statistical distribution on the data. Estimating various parameters for fitting a certain time series model, such as Autoregressive (AR), Autoregressive Moving Average (ARMA), and Autoregressive Integrated Moving Average (ARIMA) are common strategies for modeling sequential data. The Autoregressive Integral Moving Average (ARIMA) model is one of the most important and commonly used time series models. The ARIMA model's popularity stems from its statistical features as well as the well-known Box–Jenkins model-building methodology. (Zhang, 2003). Auto-Regressive Integrated Moving Average (ARIMA) is a popular forecasting method among academics (as a benchmark) and practitioners, with various applications in supply chain management (Gilbert, 2005, as cited in Punia et al., 2020). ARIMA is a popular univariate time series model that is commonly used to solve time series forecasting difficulties. Least square support vector regression (LSSVR) models for dealing with multivariate regression data that are accurate (Pai & Liu, 2018, as cited in Punia et al., 2020).

Time series forecasting has traditionally been conducted using approaches such as exponential smoothing, decision trees, Random forest, and Autoregressive Integrated Moving Average models (ARIMA). However, such methods have disadvantages, as the inability to handle faulty or missing data, and the fact that they only operate with univariate inputs (Golla, 2019). Weng et al. (2019) states that linear models, such as the multivariate linear regression model build prediction models by looking at the relationship between influencing factors and supply chain demand, and then it fits the trend of supply chain in the future. Its features include excellent modeling efficiency and ease of implementation. However, assuming a linear relationship between impact variables and supply chain demand is challenging to characterize the complex supply chain of modern organizations. Moreover, some researchers consider that historical supply chain sales data as time series, and model supply chain sales using the mobile average model and the autoregressive moving average model (Wang et al., 2010, as cited in Weng et al., 2019), which have improved performance. In sales forecasting, statistical time series analysis tools like ARIMA and SARIMA (Box et al., 2008, as cited in Weng et al., 2019) are commonly used. These methods are simple and easy to implement, and the results can be computed rapidly because they use a closed form expression for forecasting. However, the linear model does not correspond to the actual characteristics of supply chain demand change, and the anticipated results are unstable, resulting in a lack of trust in supply chain demand findings. According to Zhao et al (2017), the simplest method is the autoregressive (AR) model, which has been widely utilized in time series analysis. This strategy, however, has the issue of requiring the time series to satisfy the stationary assumption, which is usually violated in practice. Auto Regressive Moving Average (ARMA) model makes predictions based on the signal's previous values (Box & Reinsel, 1994, as cited in Mouraud, 2017). In that way, the goal of a so-called "data-driven" technique is to anticipate future values of a signal as a time series without using any other information as input. Although ARMA model does not make any physical assumptions about the underlying process, it does make some assumptions about the process itself (Mauraud, 2017). ARIMAX is the natural extension of ARIMA when explanatory variables are available in business (Dellino et al., 2018, as cited in Punia et al., 2020). Relationships can be linear or non-linear in forecasting problems (Sugihara & May, 1990, as cited in Punia et al., 2020). ARIMAX variants are adequate for linear interactions (Pai & Lin, 2005, as cited in Punia et al., 2020), but non-linear interactions are significantly more difficult.

ARIMA and other widely used statistical time series forecasting methods suppose that the time series only has linear components. However, nonlinear components are present in most real-world time series data. Several nonlinear statistical approaches have been created to handle forecasting of time series with nonlinear patterns, such as the Autoregressive Conditional Heteroscedastic (ARCH) model and the Generic Autoregressive Conditional Heteroscedastic (GARCH) model (Khashei & Bijari, 2011).

However, there are numerous varieties of these models, each of which is best suited to simulating a certain type of nonlinearity. As a result, the process of determining a suitable generic time series model becomes more difficult (Abbasimehr et al., 2020). The time series of final consumer demand cannot be adequately represented by traditional linear approaches due to the complicated nature of nonlinear patterns in their structures. Nonlinear models should be conducted when linear models fail to perform well in both training (in-sample fitting) and testing (out-of-sample forecasting) (Pacella & Papadia, 2020). Non-linear models, in most circumstances, are only capable of capturing certain types of non-linearity and do not provide generic models (Wu, 2010, as cited in Punia et al., 2020).

By considering disadvantages of classical methods and power of contemporary methods, Livieris et al. (2020) states that whereas statistical methods require assumptions such as stationarity and linear correlation between historical data, the more powerful machine learning algorithms appear to fail to recognize and capture the nonlinear and complicated behavior of gold price time series. As a result, none of these approaches can ensure the improvement of an accurate and robust forecasting model. Traditional approaches' predicting accuracies are limited, especially in non-linear and non-stationary datasets, since they are difficult to extract enough time series features. (Shen et al., 2020).

In addition to classical forecasting methods, Deep Neural Networks (DNNs) have shown a tremendous ability to map complicated non-linear feature interactions, unlike traditional statistical-based models that can only represent linear correlations in data (Reyes & Ventura, 2019). Pacella & Papadia (2020) states that deep learning techniques, unlike statistical linear models, allow arbitrary non-linear approximation functions to be learned directly from data. This enhanced generality increases the likelihood of more accurate forecasting. Because Artificial Neural Networks (ANN) can capture the many non-linearities in the data, ANNs have become one of the most popular forecasting methods. (Khashei & Bijari, 2011). In non-linear sequence learning issues, deep learning neural networks have recently showed promising outcomes. Recently, computational intelligence techniques including artificial neural networks (ANN), support vector machine (SVM), K-nearest neighbors (KNN), and adaptive neuro-fuzzy inference system (ANFIS) have been frequently used for the problem of time series prediction (Abbasimehr et al., 2020). Deep learning is a new field of machine learning research that employs Deep Neural networks (DNN) to construct artificially intelligent models. RNNs and LSTM networks, for example, are two of the most widely used deep learning approaches, and they outperformed common machine learning approaches for time-series forecasting. (Fischer & Krauss, 2018, as cited in Punia et al., 2020).

Recurrent Neural Networks (RNNs) are suited to modeling and analysis of time series data. There are several types of RNN-based models. The majority of these RNN-based models differ mostly in their ability to memorize input data. In general, a standard RNN is incapable of remembering previous data. These models are feed forwarding-based learning methods, according to deep learning jargon. Long Short-Term Memory (LSTM) networks are a specific sort of RNN model that simulates the associations between longer input and output data. (Namini et al., 2019). RNNs and LSTM networks are two of the most popular deep learning algorithms for time-series forecasting, outperforming conventional machine learning methods (Fischer & Krauss, 2018, as cited in Punia et al., 2020). Unlike other neural networks, RNN and LSTM networks have the property of maintaining information over time steps (Hochreiter & Schmidhuber, 1997, as cited in Punia et al., 2020). Recurrent Neural Networks (RNNs) are one type of ANN. Unlike feedforward ANNs, an RNNs connections between nodes form a cycle that allows signals to travel in several directions (Parmezan et al., 2019, as cited in Abbasimehr et al., 2020). By recording the activations from each time step, RNNs give a short-term memory. As a result, it is a good strategy for dealing with sequence data (Parmezan et al., 2019, as cited in Abbasimehr et al., 2020). The vanishing gradient problem and exploding gradient problem are a shortcoming of RNNs, making them difficult to train. (Bengio et al., 1994, as cited in Abbasimehr et al., 2020). The prevalent solution to overcome this weakness is to use gated architectures such as LSTM (Hochreiter & Schmidhuber, 1997, as cited in Punia et al., 2020), which can take advantage of longer-range timing information. (Wu et al., 2018, as cited in Abbasimehr et al., 2020]. Because the forecast is based on historical sales data and is a time series prediction, having memory recordings of previous data would be quite helpful. This refer that something like memory-associated networks is required. As a result, for a time series sales prediction, LSTM is the best solution. (Kishore et al., 2017). Long-Short-Term Memory (LSTM) models have the ability to recognize both long-term and short-term correlations between data components, which is critical for demand forecasting (Brownlee, 2019, as cited in Golla, 2020). According to Golla (2019), LSTM is also utilized to solve time-series problems because of its ability to learn long-term correlations in a sequence. Due to the internal memory mechanism, recurrent neural networks are very effective in identifying dependencies in sequence data. The LSTM, in particular, performs well on sequence data with long-term dependencies. Moreover, without any prior knowledge of temporal lag, LSTM proven to be successful for short-term prediction. The LSTM dynamically selects the correct time delays, demonstrating greater generalization capacity. (Tian & Pa, 2015). According to Unlu (2019), LSTM, unlike standard neural networks, is well adapted to remembering what crucial things were learned from earlier encounters. A study that looked into how well LSTM could spot patterns in multivariate time series of clinical measurements. (Lipton et al., 2015, as cited in Xue et al., 2019). Long Short-Term Memory (LSTM) is a type of RNN that was first used to deal with long input sequences in order to increase the network's ability to remember past network states and capture longer-term relationships. The bidirectional LSTM (BiLSTM) is another type of RNN. The input sequences before and after it can be used to fully use all input data in order to get the optimum learning process performance (Pacella & Papadia, 2020). Furthermore, because upgraded versions of LSTM networks, such as Graves & Schmidhuber's (2005) full gradient version, resolve the problem of vanishing gradient, LSTM networks are a good candidate for non-linear sequence forecasting. This update of LSTM networks allowed them to store information across long time steps, allowing them to be used for sequence learning. (Graves & Schmidhuber, 2005, as cited in Punia et al., 2020).

3. METHODOLOGY

This section will cover the research methodology to be used, data acquisition and selection, identification of models, model selection and training, model evaluation, and model results comparison. As a result of completing the stages defined above in this section, it is aimed to find answers to the research questions defined in the Introduction section.

Firstly, we can start with the selection and elaboration of the research methodology. Quantitative experimental research methodology will be used for this study since it is aimed to determine the best estimator of a single data set, which is the subject of the research, with different models. While doing this, numerical methods are used in the input preprocessing phase, model building and model evaluation phase. In addition, external variables that helps in the prediction of a single variable will be taken into account while establishing and training the model and they will play a role in increasing the representative power of the model.

The next subsections will continue as follows: Data acquisition & selection, data preprocessing, model definition, model building, model evaluation and assumptions & limitations.

3.1 Data Acquisition & Selection

The related study will include sales forecasting of a product group in a defense industry company. The relevant data was obtained through the Enterprise Resource Planning (ERP) of the company and its structure was modified in accordance with the principle of confidentiality. Since the notifications of customer requests are on a monthly basis, we will use the month as the time series unit time. There are two main product groups in the defense industry company. The period between the sales of the product groups belonging to the first of these groups is long and includes products that are difficult to produce. In addition, the sales of the products belonging to this group are mostly contractual and demand management is not performed with sales forecasting. On the other hand, the demand intervals of the products in the second group are more frequent, these products are relatively easier to produce, and the sales of this product group continue for a long time compared to the first group. Furthermore, demand management becomes inevitable considering the high order quantity of the second group and the demand for a new order in a short time in order to meet the customer's needs in a timely manner. According to the sales forecasting literature, long-term input, and frequent sales, which are the presence of input per unit time (day, month, year ...) are the main factors that increase the prediction power. In the light of these reasons, it was decided to conduct the study by making use of the sales data of the second group products. All sales of second group products from 2005 to 2021 form the dataset on a monthly basis.

3.2 Data Preprocessing

In time series estimation, the best prediction model is selected as a result of comparing traditional methods with contemporary methods. The models considered for comparison are ARIMA as the traditional model, Support Vector Machines (SVM) as the machine learning model, and RNN, LSTM and Convolutional Neural Network (CNN) as the deep learning methods that refers to contemporary methods. Relevant models expect an input structure suitable for their own models, rather than being able to take raw data and produce results. By considering ARIMA, data transformation is frequently required to make the time series stationary. Stationarity is a requirement for developing an ARIMA model that can be used for forecasting. The statistical features of a stationary time series, such as the mean and autocorrelation structure, remain unchanged through time. Before fitting an ARIMA model to an observed time series with trend and heteroscedasticity, differencing and power transformation are frequently used to eliminate the trend and stabilize the variance (Zhang, 2003). By considering SVM, scaling the inputs to set interval [-1,1] or [0, 1] are popular choices since there will be no bias towards specific inputs that have large values. Moreover, scaling plays an important role for getting desired accuracy level. In addition to traditional method and machine learning model, deep learning models are required for data preprocessing such that data cleaning and normalization are two main data preprocessing steps for forecasting task. If the time series contains both noisy and missing data, the noisy values are smoothed and the missing values are replaced with a suitable methodology regarding data cleaning. Normalization is a rescaling of the data from the original range so that all values are within the range of 0 and 1 (Abbasimehr et al., 2020).

3.3 Model Definition

In this section, general information will be given about the models to be used in prediction. In Autoregressive Integrated Moving Average (ARIMA) model, the future value of a variable is supposed to be a linear function of historical observations and random errors. According to Zhang (2003), the time series are generated via a process that has the following form.

$$y_t = \theta_0 + \phi_1 y_{t-1} + \phi_2 y_{t-2} + \cdots + \phi_p y_{t-p}$$
$$+ \varepsilon_t - \theta_1 \varepsilon_{t-1} - \theta_2 \varepsilon_{t-2} - \cdots - \theta_q \varepsilon_{t-q},$$

Figure 1: ARIMA Model Representation

where y_t and ε_t are the actual value and random error at time period t, respectively; ϕ_i (i = 1, 2, ..., p) and Θ_j (j = 1, 2, ..., q) are model parameters. p and q are integers and often referred to as orders of the model. Random errors, ε_t are assumed to be independently and identically distributed with a mean of zero and a constant variance of σ^2. By the help of ARIMA model representation, special cases of the ARIMA can be observed in a way that if q = 0, then the equation becomes Autoregressive (AR) model of order p. When p = 0, the model reduces to Moving Average (MA) model of order q. The primary goal of ARIMA model construction is to establish the appropriate model order (p, q). Model identification, parameter estimation, and diagnostic checking are three iterative processes in the Box–Jenkins approach. Data transformation is frequently required during the identification process to make the time series stationary. Estimation of model parameters is straightforward once a tentative model has been specified. The parameters are chosen so that the total measure of errors is as low as possible. A nonlinear optimization approach can be used to achieve this. The diagnostic testing of model appropriateness is the final phase in the model building process. This is essentially a check to see whether the model assumptions about errors are met. If the model is inadequate, a new tentative model should be identified, and the stages of parameter estimation and model verification should be repeated. This three-step model-building method is usually repeated multiple times before a good model is chosen. The final model chosen can then be utilized to make predictions.

Support Vector Machine (SVM) is a machine learning method for tackling classification and regression problems that is based on statistical learning theory. An SVM chooses a hyperplane that separates the data points with the greatest margin given a set of data points from two classes. SVM uses kernels such as linear, polynomial, and Radial Basis Function (RBF) to transfer data points into a higher-dimensional space in which they become separable for data that are not linearly separable (Han et al., 2011). The method is first designed for a linearly separable binary classification problem. Then, as Support Vector Regression (SVR), it is extended to be used for regression problems (Drucker et al., 1997). Although even the quickest SVMs have a long training period, their main qualities are extremely accurate, and their ability to simulate complex and nonlinear decision limits is quite useful. Compared to other approaches, they are far less prone to overfitting. SVM can also be used to describe the learned model in a very concise manner. Moreover, SVM has been applied in different variety of areas, especially on time series and financial prediction problems; handwritten digit recognition, speaker identification, object recognition, convex quadratic programming, and choices of loss functions are some of them (Han et al., 2013).

As an extension of RNN, LSTM is very good at forecasting time series data. The main idea of LSTM is that it introduces self-loops to produce paths where the gradient can flow for long durations that means LSTM allows information to be carried over long periods without being lost. The key distinction between an RNN and an LSTM is that the LSTM may hold long-range time dependency information and map input and output data appropriately (Greff, 2017). The LSTM network differs from traditional perceptron architecture in that it has a cell and gates that govern information flow. An input gate, a forget gate, internal state (cell memory), and an output gate are all part of the LSTM as illustrated in Figure 2.

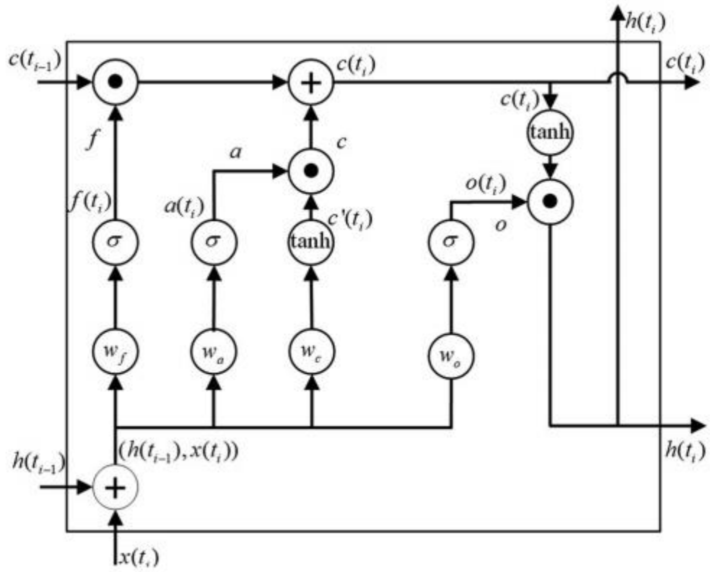

Figure 2: The Structure of LSTM

Overall, the LSTM learns by following the steps. The first step is about computing the LSTM output regarding forward learning. Then, computing the error between the resulted data and input data of each layer and the error is reversely propagated to input gate, cell, and forget gate. Lastly, Based on the error term, the weight of each gate is updated using an optimization algorithm. The four-step procedure is performed for a specified number of iterations until the optimal weights and biases are found.

General information has been given about the models to be used in time series forecasting. From now on, the methodological steps involved in time series prediction will be discussed.

3.4 Model Building

Before proceeding to the methodological steps, it will continue with the definition of the concepts that will be included while describing the steps. The first of these is sequence length. Regarding transforming data into supervised learning, time series data are reshaped into a set of instances with predefined input features and an output feature. The time series is converted into a set of input and output format. This data reshaping task is required to employ an LSTM. Sequence length is the parameter that determines how many inputs will be used in determining the labels. For example, if sequence length is 28 means that the 29^{th} days in the time series is used as a label, the previous 28 days are used as input. In the following case in Figure 3, we have sequences of 28 days, that will use to predict the next day. The figure below visualizes the what the sequence length means.

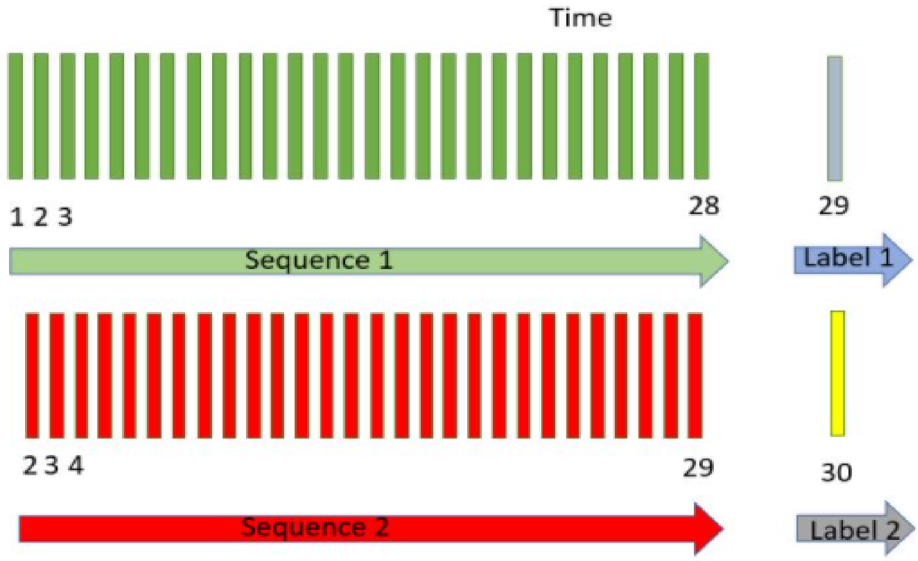

Figure 3: Illustration of Sequence Length

In our study, sequence length depends upon the domain knowledge. In our dataset, it is convenient to use a sequence length of 3,6,9 and 12 since we have monthly data and there are four quarters in a year that means considering that the financial results of the company are announced quarterly and the budgets of the customers are annual, it possible to assign the sequence length from 3 to 12 every 3 months.

After transforming data into supervised learning by using sequence length, Before model training, it is necessary to focus on the parameters of the LSTM model, which is the main study subject. The parameters are sequence length, the number of hidden layers, the number of neurons, drop rate, learning rate, batch size and epoch size. The sequence length parameter has a substantial impact on time series forecasting performance (Ribeiro et al., 2011). As a result, it is critical to test a model's performance with various lag sizes. The number of hidden layers is shown that deep neural network architectures have better generalization than single layer architectures (Hermans & Schrauwen, 2013). As a result, it is important to investigate the performance of a model with more than one layer. The task of determining the ideal number of neurons for each layer in the LSTM network is not easy. The LSTM will not be able to memorize all of the necessary information to make optimal prediction if the number of neurons is very small. Furthermore, if the number of neurons is excessively large, the LSTM will be overfit on the training instances and will not be able to effectively predict the test set (Reimers & Gurevych, 2017). Applying a dropout rate to an LSTM improves the model generalization, which improves its performance. The learning rate is a critical hyperparameter that controls how much the model weights change. Choosing the optimal learning rate is critical for producing a high-performing model (Reimers & Gurevych, 2017). Another element that influences the performance of the LSTM model is batch size. As a result, determining the ideal batch size is critical. A single iteration across all training examples is referred to as a single training epoch. The model will not capture the patterns of training examples if the number of training epochs is too short. In addition, if the epoch number is too large, the model will overfit. As a result, selecting an appropriate epoch number is critical for producing a high-performing model. A list of parameter combinations is generated based on the number of values for each hyperparameter and is used for data transformation, model building, and training. Except for the lag size parameter, the LSTM network is configured and trained using the following hyperparameters: number of hidden layers, epoch size, batch size, number of neurons, learning rate, and dropout rate.

ARIMA, SVM and LSTM will be trained with sales of the selected product group on four sequence length options that are 3, 6, 9 and 12 months. On each round, an input size choice is chosen initially, and the models are then trained using sales data for this input sequence length for the time period chosen. Following the training, the prediction phase begins. Through the chosen time frame, each model is tested to forecast four possible future dates (3, 6, 9, and 12 months).

3.5 Model Evaluation

After establishing the models in different approaches, performance metrics are needed to compare the models with each other. Performance metrics are basically measures of how close the predicted value to the true value is, so when the success of the time series prediction is tested, the values of the relevant performance metrics are expected to be low. The first performance metric planned to be used in this study is the Mean Squared Logarithmic Error (MSLE). Mean squared logarithmic error (MSLE) can be interpreted as a measure of the ratio between the true and predicted values. The mean squared logarithmic error is a variant of the Mean Squared Error, as the name implies. MSLE only concerns about the relative difference between the true and predicted values, or in other words, the percentual difference between them, thanks to the existence of the logarithm. This means that MSLE will treat small differences between true and predicted values in the same way as large differences between true and predicted values are treated. In other words, when we are willing to large errors to be significantly more penalized than small ones, in those cases where the range of the target value is large, MSLE will be good choice (Chathurika et al., 2019). It was previously stated that the data set that was the subject of the study had a fluctuating sales data, that is, serious deviations were observed even between the months of the same year. The large range value between the extreme points in the data and the fact that the MSLE works accordingly show the compatibility of the selected performance metric to the study.

Another important characteristic of MSLE is that MSLE penalizes underestimates more than overestimates (Raj, 2020). This is important for our study regarding domain knowledge such that considering the defense industry environment, underestimation of customer demand is penalized more than overestimation of customer demand since customer demand is not met in the desired amount and therefore delay penalties may arise in projects. Although overestimating has negative effects such as increase in inventories, customer penalties are incomparably greater than inventory costs. The compatibility of MSLE's penalty mechanism with domain knowledge in defense industry makes it meaningful to use MSLE in the study.

The loss is the average of the squared differences between the log-transformed true and predicted values. MSLE can be represented as in Figure 4.

$$L(y, \hat{y}) = \frac{1}{N} \sum_{i=0}^{N} (\log(y_i + 1) - \log(\hat{y}_i + 1))^2$$

Figure 4: MSLE Formula

In this study, the forecast accuracy is measured using the MSLE loss function. The least amount of loss indicates that the model is better. After calculating the performance metric for each model, considering the time frames options (3, 6, 9, and 12 months) and different types of models (ARIMA, SVM and LSTM), 4x3 evaluation matrix emerges. The comparison between the methods is made according to the performance metric values.

3.6 Assumptions & Limitations

The data set used in this study was obtained through the ERP system owned by the defense industry company. The ERP system records all transactions made in the system instantly and it provides traceability in this way that refers to the ERP strengthens the accuracy of the data in the system. In the light of this information, it is assumed that the data, which is the input of the study, is the same as the reality.

In the previous sections, it was stated that the customer's order time unit is the month. Although 17 years of sales data seems sufficient in terms of dataset size, having sales on a monthly basis means approximately 200 sales points. The data size is likely to create a limitation on the models' learning of time series structure. In order to prevent this situation, it is aimed to increase the data points in the time series as much as possible.

References

Abbasimehr, H., Shabani, M., & Yousefi, M. (2020). An optimized model using LSTM network for demand forecasting. *Computers & Industrial Engineering*, 143. doi:10.1016/j.cie.2020.106435

Chathurika, W., Marino, D., & Bulut, E. (2019). *Data Driven Hourly Taxi Drop-offs Prediction using TLC Trip Record Data.* Paper presented at 12th International Conference on Human System Interaction (HSI), Richmond, USA.

Du, S., Li, T., & Horng S. J. (2019). *Time Series Forecasting using Sequence-to-Sequence Deep Learning Framework*. Paper presented at 9th International Symposium on Parallel Architectures, Algorithms and Programming (PAAP), Taipei, Taiwan.

Elsheikh, A. H., Saba, A. I., Elaziz, M. A., Lu, S., Shanmugan, S., Muthuramalingam, T., Kumar, R., Mosleh, A. O., Essa, F. A., & Shehabeldeen, T. A. (2021). Deep learning-based forecasting model for COVID-19 outbreak in Saudi Arabia. *Process Safety and Environmental Protection*, 149, 223–233. doi:10.1016/j.psep.2020.10.048

Faloutsos, C., Gasthaus, J., Januschowski, T., & Wang, Y. (2019). *Classical and Contemporary Approaches to Big Time Series Forecasting*. Paper presented at International Conference on Management of Data (SIGMOD '19), Amsterdam, Netherlands.

Golla, K. (2019). *Product Demand Forecasting Using Deep Learning* (M.S.), Lamar University. Retrieved from https://www.proquest.com/docview/2409686625/162462CACA004D15PQ/1?accountid=13014

Gooijer, J. G., & Hyndman, R. J. (2006). 25 years of time series forecasting. *International Journal of Forecasting*, 22(3), 443–473. doi:10.1016/j.ijforecast.2006.01.001

Greff, K., Srivastava, R. K., Koutník, J., Steunebrink, B. R., & Schmidhuber, J. (2017). 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.

Hajirahimi, Z., & Khashei, M. (2019). Hybrid structures in time series modeling and forecasting: A review. *Engineering Applications of Artificial Intelligence*, 86, 83–106. doi:10.1016/j.engappai.2019.08.018

Drucker, H., Burges, C., Kaufman, L., Smola, A., & Vapnik, V. (1997). Support vector regression machines. *In Advances in neural information processing systems*, pages 155–161.

Hermans, M., & Schrauwen, B. (2013). *Training and analyzing deep recurrent neural networks*. Paper presented at the Advances in neural information processing systems.

J. Han, M. Kamber, & J. Pei, *Data Mining: Concepts and Techniques*, Morgan Kaufmann Publishers, USA, 2013.

Kavaklioglu, K. (2011) Modeling and prediction of turkey electricity consumption using support vector regression. *Applied Energy*, 88(1):368–375.

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.

Kilimci, Z. H., Akyuz, A. O., Uysal, M., Akyokus, S., Uysal, M. O., Atak Bulbul, B., & Ekmis, M. A. (2019). An Improved Demand Forecasting Model Using Deep Learning Approach and Proposed Decision Integration Strategy for Supply Chain. *Complexity*, 2019. doi:10.1155/2019/9067367

Kishore, K.K., Kumar, M.S., Murthy, M.B.S. (2017) *Automatic plant monitoring system.* In: 2017 International Conference on Trends in Electronics and Informatics (ICEI), pp. 744–748.

Kochak, A., & Sharma, S. (2015). Demand Forecasting Using Neural Network for Supply Chain Management. *International Journal of Mechanical Engineering and Robotics Research*, 11(2), 96–104.

Lakshmanan, B., Raja, P. S. N. V., & Kalathiappan, V. (2019). *Sales Demand Forecasting Using LSTM*. Paper presented at 4th International Conference on Artificial Intelligence and Evolutionary Computations in Engineering Systems, Kattankulathur, India.

Livieris, I. E., Pintelas, E., & Pintelas, P. (2020). A CNN–LSTM model for gold price time-series forecasting. *Neural Computing and Applications*, 32, 17351–17360. doi:10.1007/s00521-020-04867-x

Mouraud, A. (2017). INNOVATIVE TIME SERIES FORECASTING: AUTO REGRESSIVE MOVING AVERAGE VS DEEP NETWORKS. *ENTREPRENEURSHIP AND SUSTAINABILITY ISSUES*, 4(3), 282–293. doi:10.9770/jesi.2017.4.3S(4)

Namini, S. S., Tavakoli, N., & Namin, A. S. (2020). T*he Performance of LSTM and BiLSTM in Forecasting Time Series*. Paper presented at 2019 IEEE International Conference on Big Data (Big Data), Los Angeles, USA.

Reyes, O., & Ventura, S. (2019). Performing multi-target regression via a parameter sharing-based deep network, *International Journal of Neural Systems* 29(09), p. 1950014.

Pacella, M., & Papadia, G. (2020). *Evaluation of deep learning with long short-term memory networks for time series forecasting in supply chain management.* Paper presented at 14th CIRP Conference on Intelligent Computation in Manufacturing Engineering, CIRP ICME '20, Naples, Italy.

Punia, S., Nikolopoulos, K., Singh, S. P., Madaan, J. K., & Litsiou, K. (2020). Deep learning with long short-term memory networks and random forests for demand forecasting in multi-channel retail. *International Journal of Production Research*, 58(16), 4964–4979. doi:10.1080/00207543.2020.1735666

Raj, V. (2020). CIKM AnalytiCup 2020: *COVID-19 Retweet Prediction with Personalized Attention*. Paper presented at the 29th ACM International Conference on Information and Knowledge Management.

Reimers, N., & Gurevych, I. (2017). *Optimal hyperparameters for deep lstm-networks for sequence labeling tasks*. arXiv preprint arXiv:1707.06799.

Ribeiro, G. H. T., Neto, P. S. G. d. M., Cavalcanti, G. D. C., & Tsang, I. R. (2011). *Lag selection for time series forecasting using Particle Swarm Optimization*. Paper presented at the 2011 International Joint Conference on Neural Networks.

Sagheer, A., & Kotb, M. (2019). Time Series Forecasting of Petroleum Production using Deep LSTM Recurrent Networks. *Neurocomputing*, 323, 203–213. doi:10.1016/j.neucom.2018.09.082

Shen, Z., Zhang, Y., Lu, J., Xu, J., & Xiao, G. (2020). A novel time series forecasting model with deep learning. *Neurocomputing*, 396, 302–313. doi:10.1016/j.neucom.2018.12.084

Unlu, R. (2019). A Comparative Study of Machine Learning and Deep Learning for Time Series Forecasting: A Case Study of Choosing the Best Prediction Model for Turkey Electricity Production. *SDU Fen Edebiyat Fakültesi Fen Dergisi*, 23(2), 635–646. doi:10.19113/sdufenbed.494396

Shen, W., & Wynter, L., (2012).*Real-time road traffic fusion and prediction with gps and fixed-sensor data*. Paper presented at the 15th International Conference on. IEEE, 2012, pp. 1468–1475.

Weng, T., Liu, W., & Xiao, J. (2019). Supply chain sales forecasting based on lightGBM and LSTM combination model. *Industrial Management & Data Systems*, 120(2), 265–279. doi:10.1108/IMDS-03-2019-0170

Ma, X., Tao, Z., Wang, Y., Yu, H., & Wang, Y. (2015). *Long short-term memory neural network for traffic speed prediction using remote microwave sensor data, Transportation Research Part C: Emerging Technologies*. 54 (2015) 187–197.

Xue, N., Triguero, I., Figueredo, G. P., & Silva, D. (2019). *Evolving Deep CNN-LSTMs for Inventory Time Series Prediction*. Paper presented at 2019 IEEE Congress on Evolutionary Computation (CEC), Wellington, New Zealand.

Tian, Y., & Pan, L. (2015). Predicting short-term traffic flow by long short-term memory recurrent neural network. *Proceedings of IEEE ISC2*, 2015, 153–158.

Yang, H., & Chen, Y. P. (2018). Hybrid Deep Learning and Empirical Mode Decomposition Model for Time Series Applications. *Expert Systems With Applications*, 120, 128–138. doi:10.1016/j.eswa.2018.11.019

Zhang, G. P. (2003). Time series forecasting using a hybrid ARIMA and neural network model. *Neurocomputing*, 50, 159–175. doi:10.1016/S0925-2312(01)00702-0

Zhao, B., Lu, H., Chen, S., Liu, J., & Wu, D. (2017). Convolutional neural networks for time series classification. *Journal of Systems Engineering and Electronics*, 28(1), 162–169. doi: 10.21629/JSEE.2017.01.18