Journal of Advanced Research Design 101, Issue 1 (2023) 1-20

Journal of Advanced Research Design

Journal homepage:
https://akademiabaru.com/submit/index.php/ard
ISSN: 2289-7984

Prediction of Monthly Total Sales for a Company using Deep Learning

Chee Kel Syin[1], Shahrum Shah Abdullah[1,*]

[1] Department of Electronic Systems Engineering, Malaysia-Japan International Institute of Technology, Universiti Teknologi Malaysia, 54100 Kuala Lumpur, Malaysia

ARTICLE INFO

Article history:
Received 9 October 2022
Received in revised form 18 November 2022
Accepted 27 December 2022
Available online 25 February 2023

Keywords:
Spur gear; natural fibre composite; finite element analysis (FEA); AGMA

ABSTRACT

The popularity of deep learning in predicting the monthly total sales for a company is the current research trend in artificial intelligence. Deep learning, Long Short-Term Memory (LSTM), with the ability to remember the past, has replaced the traditional, Auto-Regressive Integrated Moving Average (ARIMA) in prediction. However, whether conventional or deep learning has the best accuracy is still an unanswered question arising from advancements in computer computational power and modern machine learning algorithms. This project compares the accuracy to determine the best algorithm to forecast the company's future three months' total sales. The monthly sales of companies are collected and pre-processed. ARIMA estimator and Keras API are used to construct the models, Root Mean Square Error (RMSE) for measuring accuracy, and successfully predicted the future three months total sales using Google Colaboratory in Python. LSTM outperforms ARIMA by obtaining a percentage error ranging from 8.84 - 12.64 %, whereas ARIMA's percentage error ranges from 71.60 - 85.84 %. Deep learning as the state-of-art in predicting monthly total sales aids the business leader in making wise business plans.

1. Introduction

Good financial planning is the foundation to ensure a company grows towards success in the business field. It is vital to wisely allocate a certain amount of money for achieving either the company's short-term or long-term goals. The most important key element for financial planning is obtaining accurate prediction results, which is not a simple task [1]. By getting accurate predictions, the company can produce the products according to the market demand, wisely invest, identify and solve potential issues efficiently, and improve the sales process [2]. Market volatility is a cause for concern in economic areas and time series prediction recently for financial sustainability [3]. Hence, prediction accuracy must be tested when deploying using the prediction algorithms, especially prediction using regression analysis, due to its several drawbacks. The relationship between variables is assumed to have remained unchanged in regression analysis. The functional relationship between variables on certain limited data might not be suitable due to many data taken into account [4].

* *Corresponding author.*
E-mail address: shahrum@utm.my

https://doi.org/10.37934/ard.101.1.120

This project aims to determine the best monthly sales prediction algorithm that obtains the highest accuracy. Hence, the machine learning algorithm, Auto-Regressive Integrated Moving Average (ARIMA), and deep learning algorithm, Long Short-Term Memory (LSTM), are introduced to solve the prediction problem. In the ARIMA algorithm, the differencing method is used to transform the non-stationary data into stationary. In comparison, Auto Regressive Moving Average (ARMA) combines Auto-Regressive (AR) and Moving Average (MA) algorithms. LSTM can recognise the structure and data patterns like non-linearity during the time series prediction. Recently, LSTM has been widely used to predict economic and finance areas, like predicting the volatility of the S&P 500 [5].

However, either a traditional algorithm or deep learning algorithm has the best accuracy when making predictions has been arising due to the development in the computer's computational power and modern machine learning algorithms like deep learning. This research aims to investigate whether the deep learning algorithm outperforms the traditional algorithm in predicting monthly sales in terms of accuracy.

ARIMA is selected to represent the traditional prediction method, whereas LSTM represents the deep learning prediction algorithm. The reason for choosing ARIMA compared to other conventional algorithms is that ARIMA shows good accuracy and excellent prediction in predicting future results. The reason for selecting LSTM is due to the ability of LSTM to memorise the past information and train the given time series data for a longer time duration. From the studies, if a good data set is provided, it was observed that LSTM achieved around 84 - 87 % average error rate reduction compared with ARIMA. Therefore, LSTM is said to be better than ARIMA [6].

The concern regarding the inaccurate prediction of a company's monthly total sales leads to the loss of golden opportunities for the company to invest in potential growth and make wise decisions regarding strategic business planning, budgeting, and risk management [7]. The consequences of wrongly predicting the monthly sales can be seen during the COVID-19 pandemic, where 2713 small and mid-size enterprises (SMEs) were shut down before October 2020. In April 2020, the manufacturing revenue suffered a substantial decline of 33 % [8,9]. Traditional prediction algorithm, ARIMA, and the latest prediction algorithm, LSTM, are the popular algorithms used in time series prediction. ARIMA algorithm is a statistical approach that uses the differencing method to convert non-stationary data into stationary data. ARIMA is effective when the time series involves short-term prediction and only needs previous data to carry out time-series prediction [10]. Therefore, this algorithm increases the accuracy of the prediction result and retains the most minimum amount of parameters. However, the ARIMA algorithm's long-term prediction result might become a straight line and hard to forecast turning points time series. ARIMA is one of the algorithms that outperform the other time series algorithms in machine learning [6].

In these few years, the LSTM algorithm is popular in the deep learning field. LSTM algorithm is a type of Recurrent Neural Network (RNN) that is designed to deal with the vanishing gradient of RNN and is effective to use when the time series involves long-term prediction [11]. However, the LSTM algorithm's time to complete the training of the data set for deploying for real-world applications is longer [12]. The average reduction in error rates of LSTM is around 85 - 87 % compared with ARIMA and shows the dominance of LSTM [6]. Hence, it is the purpose of this work to investigate this performance comparison issue further by applying it to a real company's data of choice.

2. Literature Review

2.1 Auto-Regressive Integrated Moving Average (ARIMA)

The traditional prediction method, ARIMA, was introduced by George Box and Gwilym Jenkins in 1970, and the term Box-Jenkins model is used [13]. This method is suitable to use if there are at least 50 observations [14]. Based on the research conducted by Januar *et al.,* in the Prediction of Sugar Production using the ARIMA method, the Mean Absolute Percentage Error (MAPE) obtained by ARIMA (1,0,0), ARIMA (0,0,1) and ARIMA (1,0,1), is 17 %, 19 % and 15 % respectively [15]. The research proved that ARIMA has good accuracy because the MAPE obtained is less than 20 % [16]. ARIMA is characterised by three essential elements (p, d and q), which is the combination of AR, Integration (I) and MA, where

i. AR represents the element p that depends on the relationship between the current observation and the lagged observations.
ii. I represent the element d that stands for differencing the observations to transform the non-stationary observations to stationary observations at various times.
iii. MA represents the element q that depends on the relationship between observations and the residual error terms when the MA model is applied to the lagged observations.

The equation for AR (p) can be written as,

$$x_t = c + \sum_{i=1}^{p} \emptyset_i x_{t-i} + \epsilon_t \tag{1}$$

x_t Represents the stationary variable.
c Represents the constant.
$\emptyset_i$ Represents the autocorrelation coefficients from lags one (1) until p observations.
ϵ_t Represents the Gaussian white noise series that has a mean of zero value and variance of σ_ϵ^2.

The equation for MA (q) can be written as,

$$x_t = \mu + \sum_{i=0}^{q} \theta_i \epsilon_{t-1} \tag{2}$$

μ Represents the expectation of x_t. However, the value is assumed to be zero.
θ_i Represents the weights of the current and previous observation whereas θ_0 is assumed to be 1.
ϵ_t Represents the Gaussian white noise series with a mean of zero and variance of σ_ϵ^2.

The equation for ARIMA (p and q) can be written as,

$$x_t = c + \sum_{i=1}^{p} \emptyset_i x_{t-i} + \in_t + \sum_{i=0}^{q} \theta_i \epsilon_{t-i} \tag{3}$$

Here, $\emptyset_i$ and θ_i are not equal to zero and σ_ϵ^2 is larger than zero. The differencing method can transform the non-stationary data set to a stationary data set using the *I* parameter. Therefore, the ARIMA (p, d and q) can be formed.

2.2 Long Short-Term Memory (LSTM)

The deep learning algorithm, LSTM is introduced by Hochreiter and Schmidhuber in 1997 [17]. The LSTM is a type of Recurrent Neural Network (RNN) that is developed for long-term dependencies. The hidden layer of the architecture of LSTM has a structure known as the LSTM unit. Figure 1 shows the architecture of LSTM according to Atienza [18].

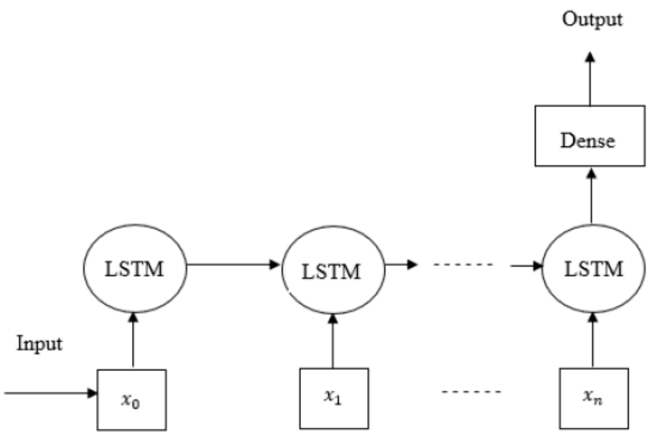


Fig. 1. Architecture of LSTM [17]

RNN contains internal memory that can function highly effectively in time series prediction. Therefore, in the LSTM structure, the prior time's data can be obtained and used the next time due to the internal memory. The LSTM can memorise the data for a long period and decide which previous data the structure wants to use.

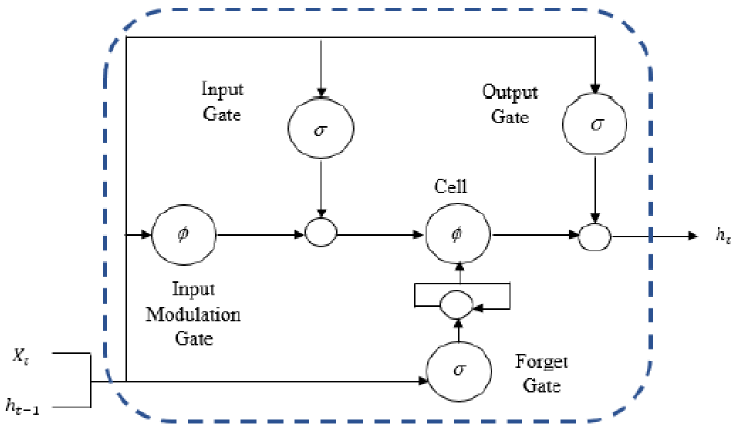


Fig. 2. Structure of LSTM [19]

X_t : Represents the input data at the t time step and the output of the previous unit.
h_t : Represents the hidden unit output.
h_{t-1} : Represents the previous output.

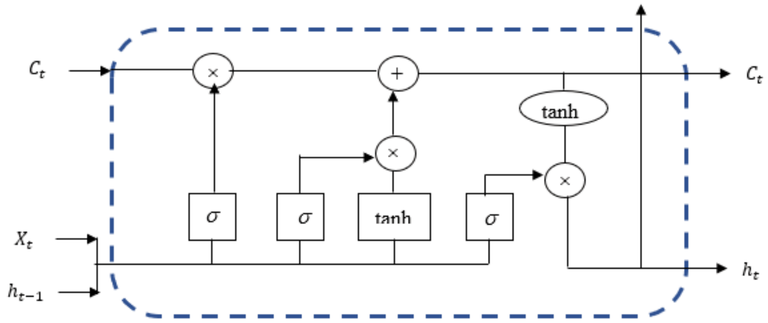


Fig. 3. Structure of LSTM [19]

C_t: Represents the cell state.
h_t : Represents the hidden state.
σ : Represents the sigmoid activation function.
tanh : Represents the tanh activation function.
$\otimes$: Represents the element-wise product
$\oplus$: Represents the concatenation operation.

Figures 2 and 3 show the structure of LSTM. Based on the research conducted by Ayse *et al.,* in 2019 about the prediction of the sales of housing in Turkey, the researchers explained the equations used in the LSTM structure as shown below [17]. The equation for the input gate, i_t^j can be written as,

$$i_t^j = \sigma(W_{xi}h_t + W_{hi}h_{t-1} + b_i)^j \tag{4}$$

The equation for the forget gate, f_t^j can be written as,

$$f_t^j = \sigma(W_{xf}x_t + W_{hf}h_{t-1} + b_f)^j \tag{5}$$

The equation for the output gate, σ_t^j can be written as,

$$\sigma_t^j = \sigma(W_{x\sigma}x_t + W_{h\sigma}h_{t-1} + b_\sigma)^j \tag{6}$$

σ : Represents the sigmoid function.
W : Represents the weight matrices.
b : Represents the voltage vectors.

The equation to update the memory cell of the LSTM unit at t time, c_t^j can be written as,

$$c_t^j = f_t^j c_{t-1}^j + i_t^j c_t^{-j} \tag{7}$$

The equation to update the new memory content, f_t^j can be written as,

$$f_t^j = tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_f)^j \tag{8}$$

The equation to calculate the output for the LSTM unit, h_t^j can be written as,

$$h_t^j = \sigma_t^j \tanh(c_t^j) \quad (9)$$

Epoch is defined as the number of iterations used to calculate the network weights when training the data set. Epoch means the data set has been forwarded back to the network. The data set is sent to the network multiple times to increase the accuracy of the prediction algorithm by updating the weights. However, the number of epochs needed for every data set is different due to the data set's different features.

3. Methodology

3.1 Project Implementation Flowchart

To ensure that developing the project can be done smoothly and systematically, a project implementation flowchart with seven parts was created. First and foremost, some past research papers are reviewed and used as references for the project. Second, it is essential to understand the theory of ARIMA and LSTM to make a comparison between both algorithms. Third, there is a need to know how to carry out the prediction system. Fourth, the coding for both ARIMA and LSTM is done using Python language. Fifth, the simulation of this project is carried out using Google Colaboratory. Sixth, if the results obtained are satisfied, proceed to the next step. Finally, the obtained results are analysed and evaluated to discover which algorithm is the best for predicting the monthly sales of a company in terms of accuracy. Figure 4 shows the flowchart on how to carry out this project.

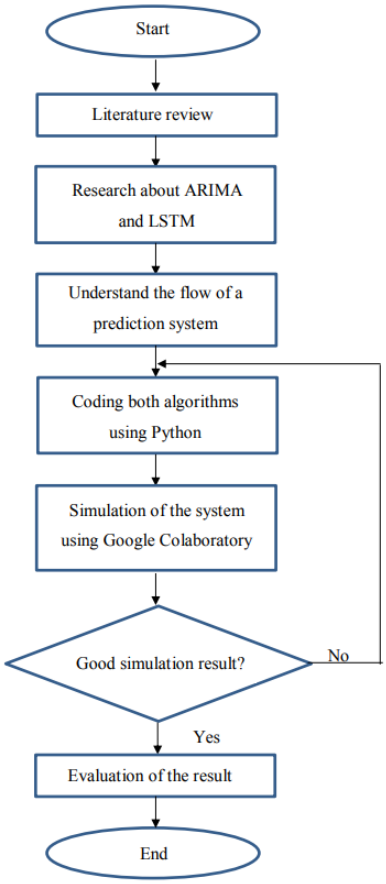


Fig. 4. Flowchart of project implementation

3.2 Block Diagram of the Proposed Procedures

Figure 5 shows the steps to carry out the ARIMA and LSTM algorithms. The proposed procedures are described in detail in the following sections.

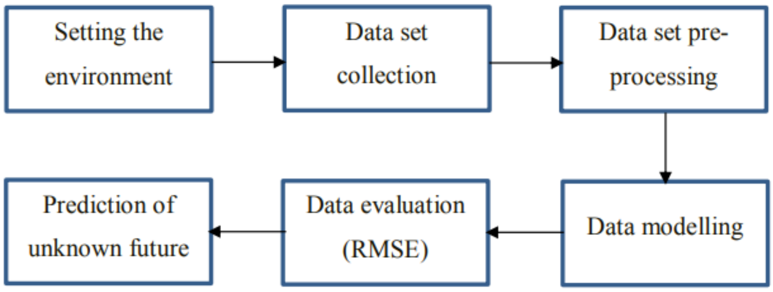


Fig. 5. Block diagram of the proposed procedures

3.3 Setting the Environment

This prediction project is a computer-based simulation that requires only Google Colaboratory as the software. Google Colaboratory is a free software that enables the programmer to write codes and execute the codes cell by cell. The software is suitable for the implementation of machine learning projects and data analysis.

In this software, the language used is Python. Python is a high-level interpreted language which uses commonly in the machine learning field. Python reads the codes line by line at a time. Therefore, through Google Colaboratory, the functionality of each code is easily understood. Python is easy to learn for a beginner as the language emphasises code readability. Table 1 shows the libraries installed in the project and the functionalities in this project are described in detail.

Table 1
Functionality of the libraries

Libraries	Functions
`pandas`	• Read the data set in the .csv file into Google Colaboratory.
`matplotlib`	• Display the result using a graph.
`statsmodels`	• Implement Augmented Dickey-Fuller (ADF) test.
	• Plot Autocorrelation Function (ACF) and Partial Autocorrelation Function (PACF) graphs.
	• Implement ARIMA prediction.
`pmdarima`	• Determine the p, d, q parameters of the ARIMA model.
`warnings`	• Filter warnings.
`sklearn`	• All the mean square error functions for Root Mean Square Error (RMSE) evaluation.
	• Call MinMaxScaler function for data normalisation and denormalisation.
`numpy`	• Convert data set into array format in LSTM.
`tensorflow`	• Call the Keras function for the LSTM model.
`math`	• Call the square root function for RMSE evaluation.

3.4 Software Flowchart for ARIMA and LSTM Algorithms
3.4.1 ARIMA

Figures 6 and 7 show the flowchart and pseudo code for the ARIMA algorithm. First and foremost, the data set of the company in the .csv file is inserted into the Google Colaboratory. Second, there is a need to pre-process the data, such as converting the dates string into time series and eliminating the empty columns. The data set is displayed in a graph for viewing. Third, there is a need to carry out the stationary test to determine whether the data set has a constant mean and variance. If the data set is stationary, proceed to the next step. If the data set is not stationary, the differential operation is carried out. Fourth, the ACF and PACF graphs are generated to obtain the p and q parameters of the ARIMA model. Fifth, the best ARIMA (p, d and q) model is determined. Sixth, the data set is split into training and testing sets with a ratio of 7: 3. Seventh, the ARIMA (p, d and q) model is ready for monthly sales prediction. Eighth, the accuracy of the ARIMA model is determined by evaluating the RMSE. If the RMSE error is satisfied, proceed to the next step. If the RMSE error is unsatisfied, return to the fifth step to change the p, d and q parameters. Ninth, the unknown future three months' monthly sales are predicted using the determined ARIMA model. Tenth, the graph of predicted monthly sales of the company is displayed.

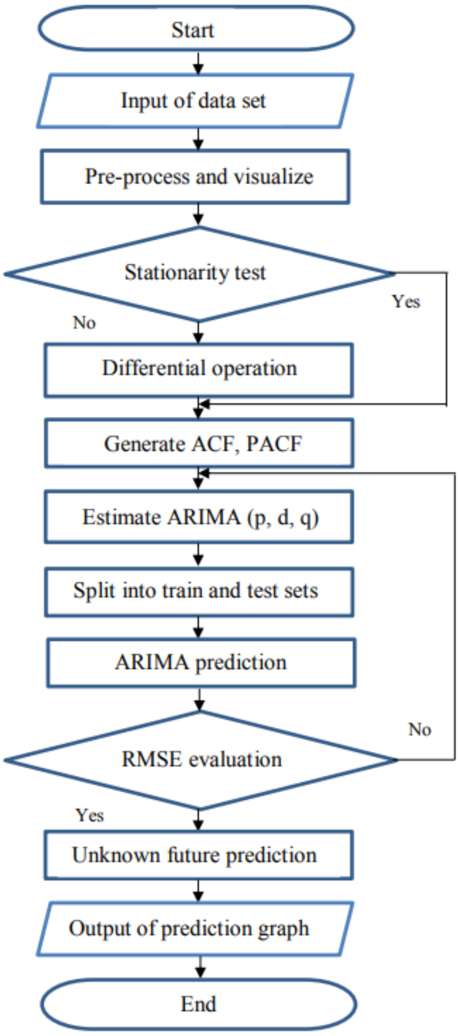


Fig. 6. Flowchart of ARIMA algorithm

1) Insert and pre-process the data set of a company
2) Visualize the data set
3) Test the stationarity of data set
4) Determine the usage of ARMA model using ACF and PACF Graphs
5) Determine the best fitting order of ARIMA (p, d, q) using pmdarima
6) Split the data set into training and testing sets (7:3)
7) Construct the ARIMA model
8) Measure ARIMA's performance
   - Measure the RMSE
   - Calculate the percentage error between RMSE and actual mean
9) Prediction of unknown future 3 months total sales
   - May 2021, June 2021, and July 2021

Fig. 7. Pseudo code for ARIMA algorithm

3.4.2 LSTM

Figures 8 and 9 show the pseudo code and flowchart for the LSTM algorithm. First and foremost, the data set of the company in the .csv file is inserted into the Google Colaboratory. Second, there is a need to pre-process the data, such as converting the dates string into time series and eliminating the empty columns. The data set is displayed in a graph for viewing. Third, there is a need to normalise the data set to the range of 0 to 1 to ease the LSTM training. Fourth, the data set is split into training and testing sets with a ratio of 7: 3. Fifth, the LSTM model is constructed. Sixth, the epochs and batch size parameters of the LSTM model are initialised. Seventh, the data set is trained using the constructed LSTM model. The monthly sales of the company are predicted. Eighth, the data set is denormalised using the scaler into the original data set scale. Ninth, the accuracy of the LSTM model is determined by evaluating the RMSE. If the RMSE error is satisfied, proceed to the next step. If the RMSE error is unsatisfied, return to the sixth step to change the epochs and batch size parameters. Tenth, the unknown future three months' monthly sales are predicted using the determined LSTM model. Eleventh, the graph of predicted monthly sales of the company is displayed.

1) Insert and pre-process the data set of a company
2) Visualize the data set
3) Reshape the data set using MinMax scaler
4) Split the data set into training and testing sets (7:3)
5) Construct the LSTM model
   - Determine the number of LSTM layers and Dense layer
   - Determine epochs
   - Determine batch sizes
6) Transform back the data set to original scale
7) Measure LSTM's performance
   - Measure the RMSE
   - Calculate the percentage error between RMSE and actual mean
8) Prediction of unknown future 3 months total sales
   - May 2021, June 2021, and July 2021

Fig. 8. Pseudo code for LSTM algorithm

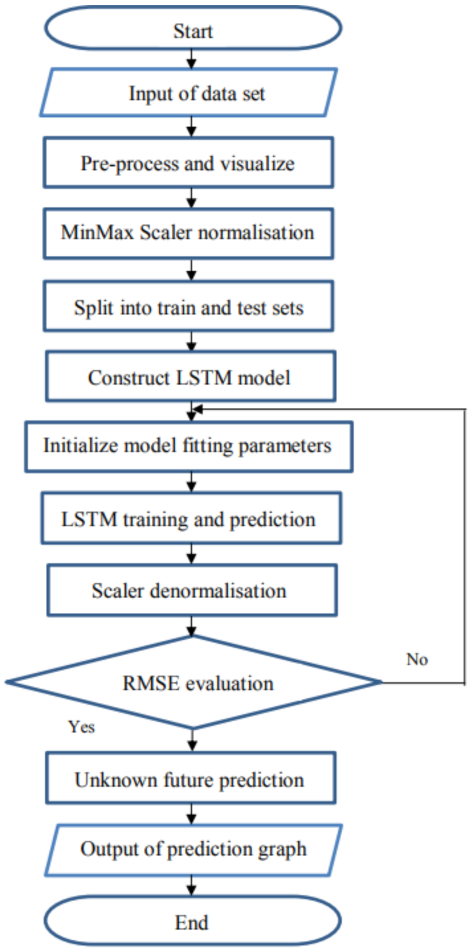


Fig. 9. Flowchart of LSTM Algorithm

4. Result and Discussion

4.1 Data Set Collection

Table 2 shows the duration of monthly sales taken for every company. For ASUSTek Computer Inc., the monthly sales taken were recorded from January 2005 to April 2021, with a total of 196 months. For Delta Electronics Inc., the monthly sales taken were recorded from January 2009 to April 2021, with a total of 148 months. For Taiwan Semiconductor Manufacturing Company, the monthly sales taken were recorded from January 2007 to April 2021, with a total of 172 months. For Quanta Computer Inc., the monthly sales taken were recorded from January 2007 to April 2021, with a total of 172 months.

The duration of monthly sales taken as a data set was different for each company. However, the last monthly sale taken for every company was until April 2021. The data set collected for every company should be as many as possible. The more the number of monthly sales observations for a company, the more accurate the predicted monthly sales. When the number of monthly sales observations collected was too less, for the ARIMA algorithm, a non-stationary time series was unable to undergo the differencing process. Thus, the ARIMA algorithm cannot be used to predict that time series. For the LSTM algorithm, the predicted results were computed perfectly even with a lesser number of monthly sales observations. This statement was supported by Boulmaiz, *et al.,* through the project experiments that conclude LSTM was better than the traditional algorithm as a statical indicator because the algorithm worked well even with only three years of data set

[20]. The three-year data set was able to obtain the identical predicted results as the nine-year data set.

Table2
Duration of monthly sales for each company

Company	Duration	Number of observations (months)
ASUSTeK Computer Inc.	January 2005 – April 2021	196
Delta Electronics Inc.	January 2009 – April 2021	148
Taiwan Semiconductor Manufacturing Company	January 2007 – April 2021	172
Quanta Computer Inc.	January 2007 – April 2021	172

4.2 Split the Time Series into Testing and Training Sets

Furthermore, the time series was split into the training and testing sets with a ratio of 7:3 for every company and shown in Table 3. The first 137 months and the last 59 months of ASUSTeK Computer Inc. were used as the training and testing set. For Delta Electronics Inc., the training and testing sets used the first 104 months and the last 44 months of the data set. The first 120 months and the last 52 months for Taiwan Semiconductor Manufacturing Company and Quanta Computer Inc. were used as the training and testing sets.

Table 3
Training and testing sets of each company

Company	Training set	Testing set
ASUSTeK Computer Inc.	137	59
Delta Electronics Inc.	104	44
Taiwan Semiconductor Manufacturing Company	120	52
Quanta Computer Inc.	120	52

4.3 Formulation of ARIMA and LSTM Models
4.3.1 ARIMA

Tables 4 and 5 show the ACF and PACF graphs, the stationarity test, and the `pmdarima` method. These steps were carried out to formulate the ARIMA (p, d and q) model for every company. For ACF and PACF graphs, the blue colour area indicate the significant threshold where the spikes that exceed the colour area were considered important. For determining the orders of p and q, both the PACF and ACF graphs will be in the damped exponential or sine function pattern as shown in the observations. Thus, the ARMA (p and q) should be used for the four targeted companies instead of AR or MA models. If the AR (p) model was used, the ACF graph should be in a geometric decay pattern, whereas the PACF graph should have significance until certain p lags. If the MA (q) model was used, the ACF graph should have significance until certain p lags and the PACF graph shows a geometric decay pattern.

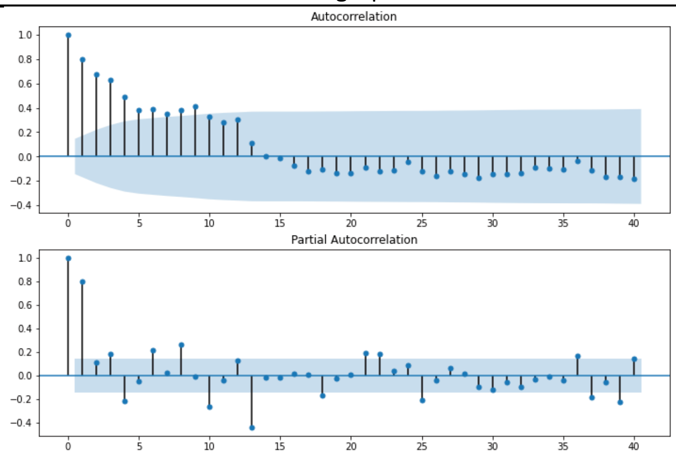
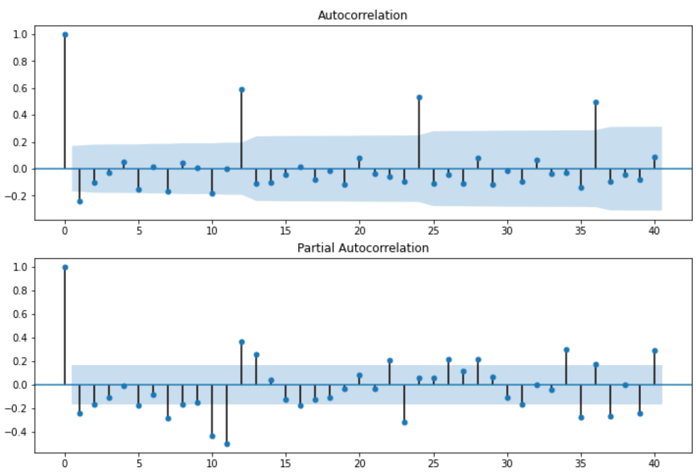
However, it is very tricky to determine the ARIMA (p, d and q) model using only ACF and PACF graphs as the graphs give many possible combinations. Thus, the `pmdarima` method was used to solve the problem. The best model was generated using the `pmdarima` method for every company. For ASUSTeK Computer Inc., the best model was ARIMA (3, 0 and 2), as the AIC value of 4034.533 was the lowest value among all the models. By comparing the order of d with the stationarity test, the time series for the company obtained a *p*-value of 0.0123 less than 0.05. Thus, proves that the data had no unit root and is stationary. Therefore, the order of d = 0 as there was no differencing operation carried out.

For Delta Electronics Inc., the best model was ARIMA (1, 1 and 1) because the AIC value of 2625.539 was the lowest value among all the models. By comparing the order of d with the stationarity test, the time series for the company obtained a *p*-value of 0.0215 less than 0.05. Thus, proves that the data had no unit root and is stationary after the first differencing operation. Therefore, the order of d = 1.

For Taiwan Semiconductor Manufacturing Company, the best model was ARIMA (2, 1 and 1) because the AIC value of 3562.320 was the lowest value among all the models. By comparing the order of d with the stationarity test, the time series for the company obtained a *p*-value of 0.0141 lesser than 0.05. Thus, proves that the data had no unit root and is stationary after the first differencing operation. Therefore, the order of d = 1.

For Quanta Computer Inc., the best model was ARIMA (2, 1 and 1) because the AIC value 3691.239 was the lowest value among all the models. By comparing the order of d with the stationarity test, the time series for the company obtained a *p*-value of 0.00000243 lesser than 0.05. Thus, proves that the data had no unit root and is stationary after the first differencing operation. Therefore, the order of d = 1.

Table 4
Table of observation of the ACF and PACF graphs

Company	Observations of ACF and PACF graphs
ASUSTeK Computer Inc.	
Delta Electronics Inc.	

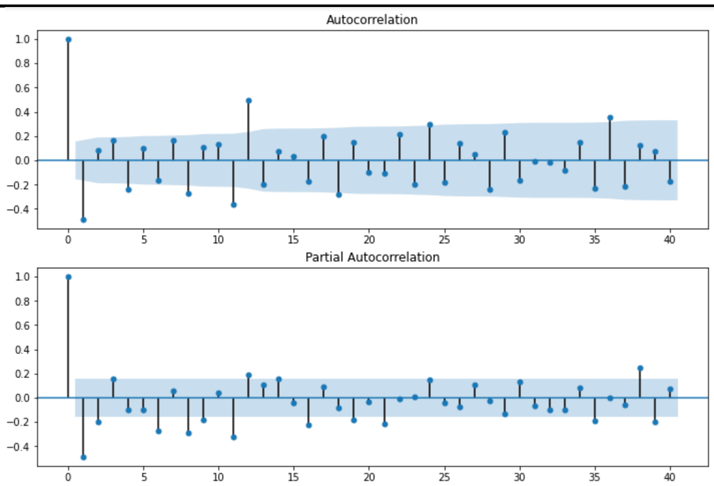
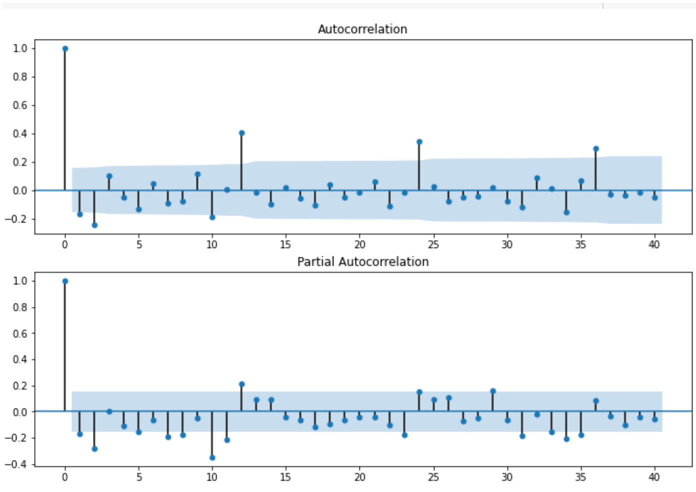
Taiwan Semiconductor Manufacturing Company	
Quanta Computer Inc.	

Table 5
ARIMA model for each company

Company	*p*-value	ARIMA (p, d and q)	AIC
ASUSTeK Computer Inc.	0.0123	ARIMA (3, 0 and 2)	4034.533
Delta Electronics Inc.	0.0215	ARIMA (1, 1 and 1)	2625.539
Taiwan Semiconductor Manufacturing Company	0.0141	ARIMA (2, 1 and 1)	3562.320
Quanta Computer Inc.	0.00000243	ARIMA (2, 1 and 1)	3691.239

4.3.2 LSTM

Figure 10 shows the LSTM model used to train the four targeted companies. Three layers of LSTM are used in this project to increase the depth of the model. The deeper the model, the higher the accuracy of the prediction results. Stacked LSTM consists of multiple LSTM layers, which are shown in Figure 11. The working mechanism of the stacked LSTM is the top LSTM layer produces a sequential output instead of a single value output for the next LSTM layer. In other meaning, there is only an output per input time step.

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
lstm (LSTM)                  (None, 10, 50)            10400
_________________________________________________________________
lstm_1 (LSTM)                (None, 10, 50)            20200
_________________________________________________________________
lstm_2 (LSTM)                (None, 50)                20200
_________________________________________________________________
dense (Dense)                (None, 1)                 51
=================================================================
Total params: 50,851
Trainable params: 50,851
Non-trainable params: 0
```

Fig. 10. LSTM model for the companies

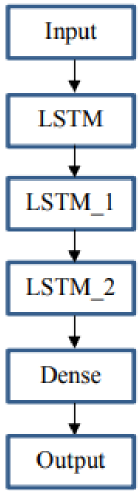


Fig. 11. LSTM model

Figure 12 shows the epochs and batch size parameters used to fit the LSTM model. The 5000 epochs are used because the more the number of epochs, the higher the accuracy of the prediction results. Compared to 1000 epochs, it is evident that the model does not have enough training as the prediction on the testing set does not fit the actual monthly sales nicely. Therefore, the epoch is increased to 5000.

The second objective is to identify and formulate ARIMA and LSTM algorithms for predicting the monthly total sales for a company. The result obtained should be that LSTM outperforms ARIMA because LSTM's accuracy is better than the ARIMA algorithm. The formulation of the algorithms for ARIMA and LSTM prediction is successful.

```
model.fit(X_train,y_train,validation_data=(X_test,ytest),epochs=5000,batch_size=64,verbose=1)
```

Fig. 12. The parameters to fit the model

4.4 Performance Evaluation of ARIMA and LSTM Algorithms

Table 6 shows the percentage error measured for ARIMA and LSTM algorithms. The actual mean column recorded the mean of the testing set for the four targeted companies. The experimental RMSE values for ARIMA and LSTM algorithms are measured. Theoretically, the experimental RMSE's result should be close to the actual mean. Thus, the percentage error is calculated to measure the extent of accuracy for both algorithms. The percentage error recorded for ASUSTeK Computer Inc. is 79.23 % for ARIMA and 8.84 % for LSTM. The percentage error recorded for Delta Electronics Inc. is 85.84 % for ARIMA and 11.32 % for LSTM. The percentage error recorded for Taiwan Semiconductor Manufacturing Company is 84.95 % for ARIMA and 12.64 % for LSTM. The percentage error recorded for Quanta Computer Inc. is 71.60 % for ARIMA and 9.34 % for LSTM.

By observing the percentage error obtained, the percentage error for the ARIMA algorithm ranged from 71.60 - 85.84 %. As for the LSTM algorithm, the percentage error ranged from 8.84 - 12.64 %. This result has proven that the deep learning algorithm, LSTM was indeed better than the traditional algorithm, ARIMA. The high percentage error for the ARIMA algorithm was due to the ARIMA algorithm unable to detect the seasonal component of the time series. When fitted the trained model to the testing set, the predicted results obtained were a rough approximation. The LSTM algorithm was the most suitable method when the algorithm requires to study of the long-term dependencies time series. LSTM can remember the previous time series. When fitted the trained model to the testing set, the predicted results fit nicely with the actual monthly sales.

Table 6
Percentage error for ARIMA and LSTM algorithms

Company	Actual mean	Experimental RMSE		Percentage error (%)	
ASUSTeK Computer Inc.	34427.92	ARIMA	LSTM	ARIMA	LSTM
Delta Electronics Inc.	22017.80	7150.55	37471.69	79.23	8.84
Taiwan Semiconductor Manufacturing Company	94074.75	3117.60	19526.06	85.84	11.32
Quanta Computer Inc.	86992.83	14158.63	82180.83	84.95	12.64

4.5 Predicted Monthly Sales of the Companies using ARIMA and LSTM

Tables 7 and 8 show the future three months of total sales for a company predicted using ARIMA and LSTM algorithms. The recorded monthly sales for May, June and July 2021 were predicted without actual monthly sales. Thus, both algorithms were formulated successfully to predict future monthly sales. The monthly sales from the LSTM prediction was suggested to be used for the company future planning instead of ARIMA prediction. LSTM prediction outperforms ARIMA prediction.

Table 7
ARIMA algorithm's predicted results

Company	ARIMA Predicted Future 3 Months		
ASUSTeK Computer Inc.	2021-05-01	2021-06-01	2021-07-01
Delta Electronics Inc.	38577	43184	36334
Taiwan Semiconductor Manufacturing Company	25500	25250	25000
Quanta Computer Inc.	114000	122500	112500

Table 8
LSTM algorithm's predicted results

Company	LSTM Predicted Future 3 Months		
ASUSTeK Computer Inc.	2021-05-01	2021-06-01	2021-07-01
Delta Electronics Inc.	36226	41731	41029
Taiwan Semiconductor Manufacturing Company	18295	17263	15186
Quanta Computer Inc.	66289	85799	91224

Figures 13 to 20 show the results of ARIMA and LSTM predictions. For the ARIMA graph, the orange line represents the actual monthly sales of a company. The blue line represents the predicted monthly sales of a company. The grey colour area in the graph shows 95 % of the confidence interval. As long as the future three-months total sales fall on the grey area within the acceptable range. The LSTM graph is the combination of actual monthly sales with future three-month total sales. The future monthly sales were trained as LSTM learns past observations and applies the knowledge to predict future values.

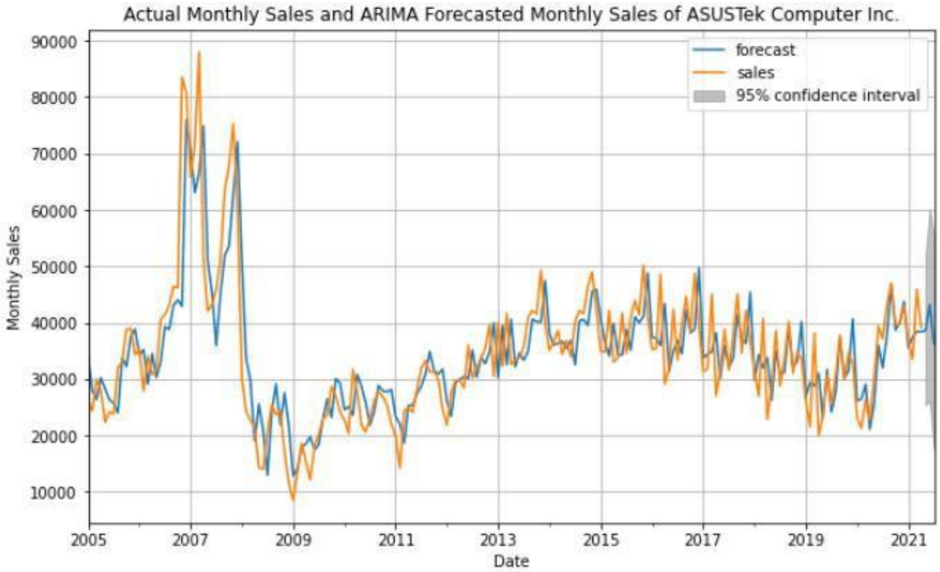


Fig. 13. Predicted result for ASUSTeK Computer Inc. using ARIMA algorithm

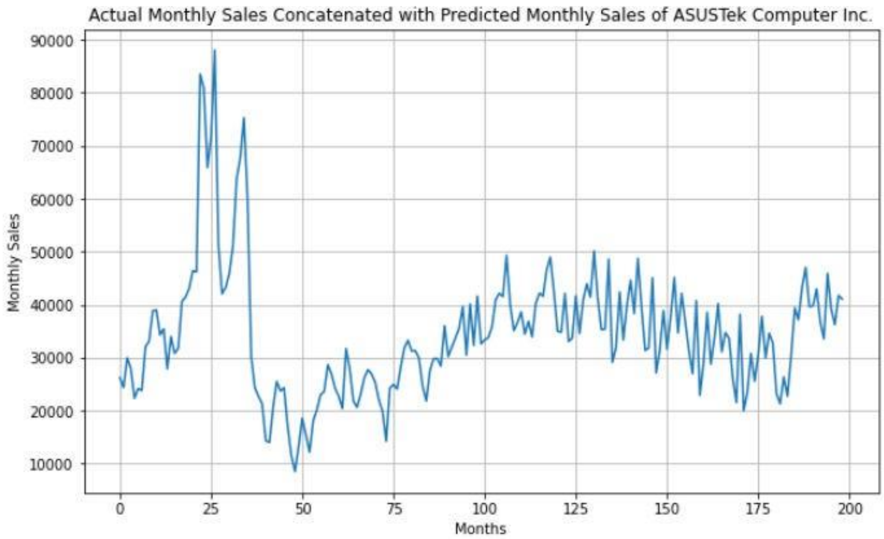


Fig. 14. Predicted result for ASUSTeK Computer Inc. using LSTM algorithm

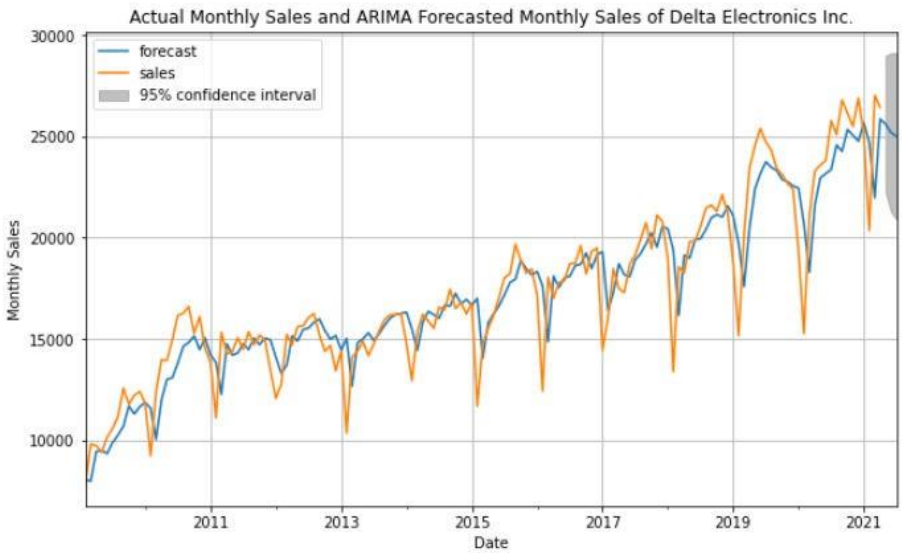


Fig. 15. Predicted result for Delta Electronics Inc. using ARIMA algorithm

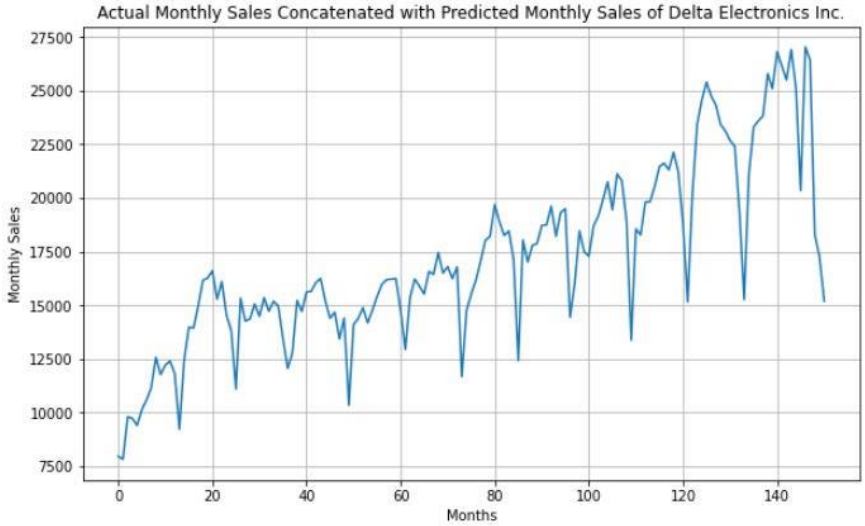


Fig. 16. Predicted result for Delta Electronics Inc. using LSTM algorithm

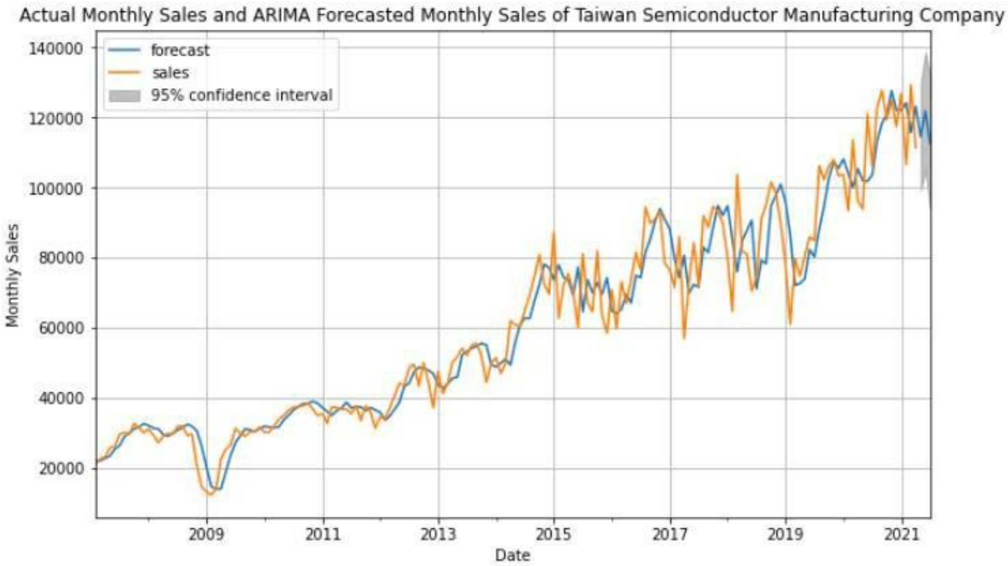


Fig. 17. Predicted result for Taiwan Semiconductor Manufacturing Company using ARIMA algorithm

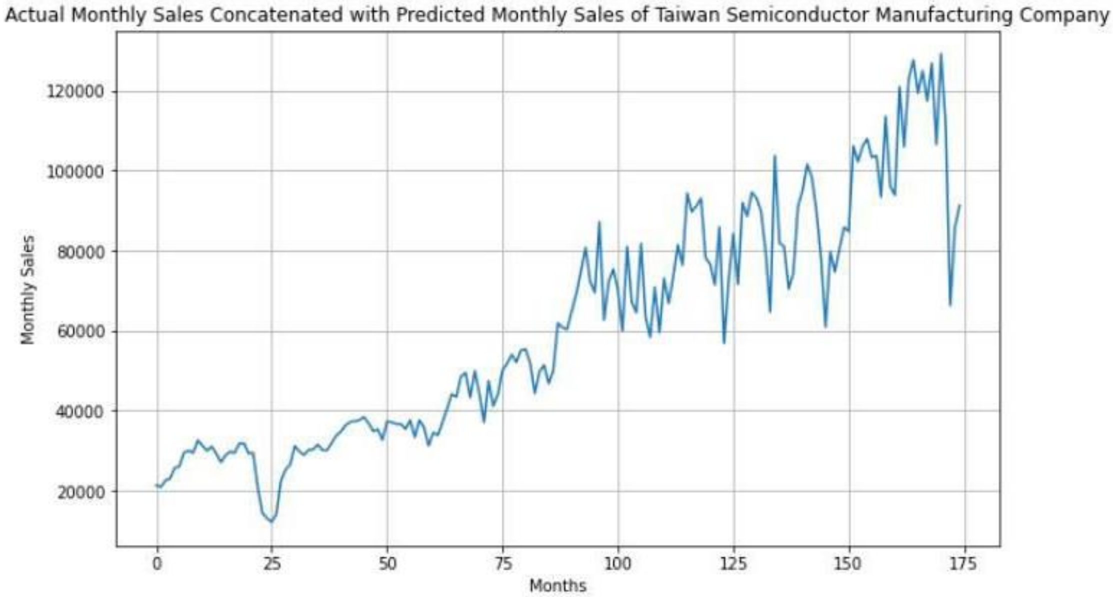


Fig. 18. Predicted result for Taiwan Semiconductor Manufacturing Company using LSTM algorithm

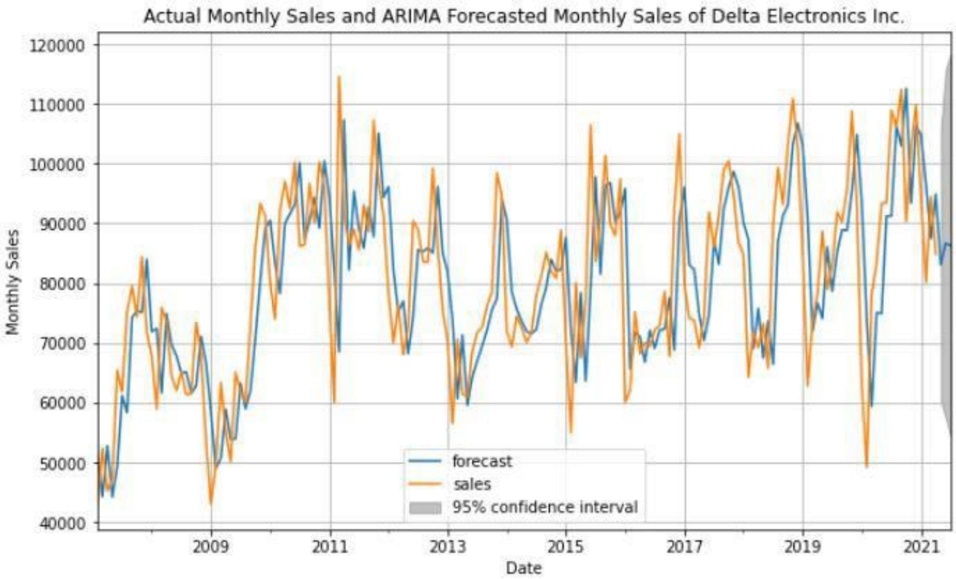


Fig. 19. Predicted result for Quanta Computer Inc. using ARIMA algorithm

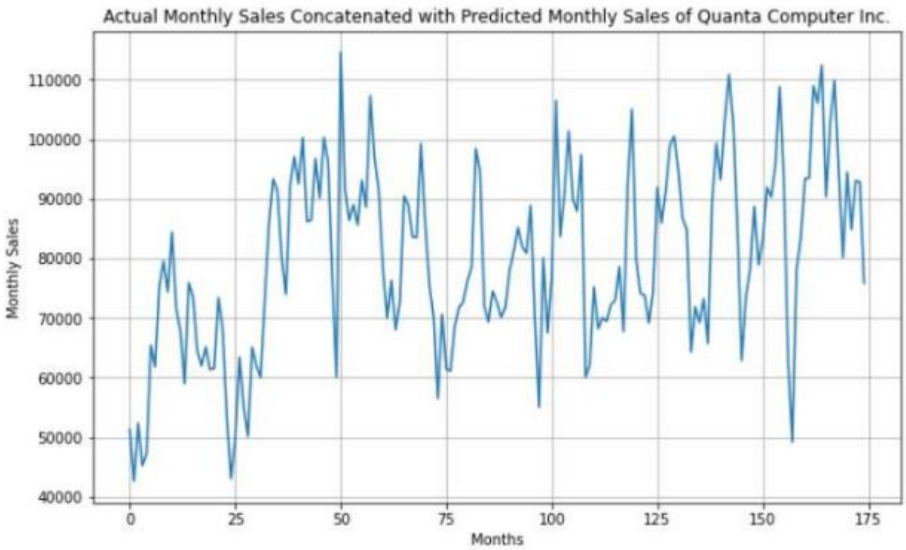


Fig. 20. Predicted result for Quanta Computer Inc. using LSTM algorithm

5. Conclusion

Four companies which are ASUSTeK Computer Inc., Delta Electronics Inc., Taiwan Semiconductor Manufacturing Company and Quanta Computer Inc. were chosen as the target companies. The duration of months for each company was collected as the data set. The data set was pre-processed before undergoing training using the model to ease the forecasting process.

Next, the deep learning algorithm (LSTM) and the traditional algorithm (ARIMA), were successfully developed using Google Colaboratory in Python. This project was successfully proven that the deep learning, LSTM algorithm outperforms the traditional algorithm, ARIMA, in predicting the monthly total sales for a company by having a percentage error as low as 8.84 % when predicting the ASUSTeK Computer Inc. company. Whereas with the traditional algorithm, ARIMA can only achieve the lowest percentage error of 71.60 % when predicting the Quanta Computer Inc. company.

Besides, both algorithms also successfully validate the future monthly sales in May, June and July of 2021 for the four targeted companies. The deep learning algorithm, LSTM, have been proven for doing strategic planning for a company. The analyses result was more accurate using a deep learning approach as LSTM can remember past values.

Acknowledgement
This research was not funded by any grant.

References
[1] The Top 5 Forecasting Problems Your Business May Face — and How to Tackle Them. Wipfli. (2020, June 5).
[2] Walker, S. *Why Is Sales Forecasting Important? 5 Reasons You Need It*. MethodData. (2020, June 8).
[3] Siami-Namini, Sima, Neda Tavakoli, and Akbar Siami Namin. "A comparison of ARIMA and LSTM in forecasting time series." In *2018 17th IEEE international conference on machine learning and applications (ICMLA)*, pp. 1394-1401. IEEE, 2018. https://doi.org/10.1109/ICMLA.2018.00227
[4] What are some of the advantages and disadvantages of regression analysis? (n.d.). Quora.
[5] Huck, Nicolas. "Pairs selection and outranking: An application to the S&P 100 index." *European Journal of Operational Research* 196, no. 2 (2009): 819-825. https://doi.org/10.1016/j.ejor.2008.03.025
[6] Sharmistha, C. *ARIMA/SARIMA vs LSTM with Ensemble learning Insights for Time Series Data*. (n.d.). Data Science Central. (2020, July 26).

[7] Mahalingam, K. *Importance of Sales Forecasting & Six factors to consider for Accurate Forecasting*. Chargebee's SaaS Dispatch. (2020b, September 14).
[8] Achariam, T. *Covid-19: 2,713 SMEs closed down between March and October 2020*. The Edge Markets. (2020, November 18).
[9] Zainul, E. *Malaysia's manufacturing sales post a significant drop of 33% in April 2020*. The Edge Markets. (2020b, June 11).
[10] O'Donovan, Thomas M. "Short term forecasting: An introduction to the Box-Jenkins approach." *(No Title)* (1983).
[11] Brownlee, J. *Time Series Prediction with LSTM Recurrent Neural Networks in Python with Keras*. Machine Learning Mastery. (2020, August 28).
[12] GeeksforGeeks. *Understanding of LSTM Networks*. (2020, May 11).
[13] Stellwagen, Eric, and Len Tashman. "ARIMA: The models of Box and Jenkins." *Foresight: The International Journal of Applied Forecasting* 30 (2013): 28-33.
[14] *The Box-Jenkins Method.* (n.d.). Retrieved from https://ncss-wpengine.netdna-ssl.com/wp content/themes/ncss/pdf/Procedures/NCSS/The_Box-Jenkins_Method.pdf
[15] Putra, Januar Adi, Faishal Basbeth, and Saiful Bukhori. "Sugar production forecasting system in PTPN XI semboro jember using autoregressive integrated moving average (ARIMA) method." In *2019 6th International Conference on Electrical Engineering, Computer Science and Informatics (EECSI)*, pp. 448-453. IEEE, 2019. https://doi.org/10.23919/EECSI48112.2019.8977010
[16] Kristiean Margi, S. P. Analysis and Application of Single Exponential Smoothing Method for Prediction of Sales in Specific Periods (Case Study: PT. Media Cemara Kreasi). *Prosiding SNATIF.* (2015).
[17] Temur, Ayse Soy, Melek Akgün, and Gunay Temur. "Predicting housing sales in Turkey using ARIMA, LSTM and hybrid models." (2019). https://doi.org/10.3846/jbem.2019.10190
[18] Atienza, R. *LSTM by Example using Tensorflow - Towards Data Science*. Medium. (2018, June 21).
[19] LlSourcell. (n.d.). LlSourcell/LSTM_Networks. https://github.com/llSourcell/LSTM_Networks/blob/master/LSTM Demo.ipynb
[20] Boulmaiz, T., Mawloud Guermoui, and Hamouda Boutaghane. "Impact of training data size on the LSTM performances for rainfall–runoff modeling." *Modeling Earth Systems and Environment* 6 (2020): 2153-2164. https://doi.org/10.1007/s40808-020-00830-w