e-ISSN: 2614-8404
p-ISSN: 2776-3234

PREDICTION OF INCOMING ORDERS USING THE LONG SHORT-TERM MEMORY METHOD AT PT. XYZ

Lukman Irawan[1], Fauzi[2], Denny Andwiyan[3]
[1] Master Program in Computer Science, Faculty of Information and Technology, Budi Luhur University
[2] Master Program in Computer Science, Faculty of Information and Technology, Budi Luhur University
[3]Science and Technology Faculty, Information System Department, Raharja University
email: [1]lukman.irawan26@gmail.com, [2]fauzi.said25@gmail.com, [3]andwiyan@raharja.info

Abstract – **Currently the need for domestic packaging paper continues to increase, driven by the level of consumer awareness about sustainable packaging. PT XYZ is a local company engaged in the Corrugated Cardboard Box (KKG) industry. So far, the problems in fulfilling incoming orders every month are not optimal with an average of about 30% inaccuracy. This is because the orders that enter cannot be predicted. As an effort to win market competition in packaging paper, PT. XYZ must improve the fulfillment of incoming orders by predicting incoming orders using the Long Short-Term Memory (LSTM) method. The aim of this research is to provide a predictive model for incoming orders in accordance with the needs of order fulfillment to be applied to production planning. So that order fulfillment can be on time. The method used in predicting incoming orders is the Long Short-Term Memory (LSTM) method using weighting evaluations with the lowest Root Mean Squared Error (RMSE) and Augmented Dickey-Fuller test (ADF). The test results of the LSTM method with parameter sizes of Batch: 1 Epochs: 5000 Neurons: 1 show that the RMSE for MDM products is 8.767582 and 0.287924, LNR products are 10.623984 and 0.466621, WTP products are 1.636849 and 0.361515 lower than the size of the fit parameters for other LSTM models, and the ADF Statistic value for MDM products -6.137597, LNR -6.753697, WTP -4.872927**.

Keywords* – Prediction, Planning, LSTM, *Time Series.

I. INTRODUCTION

Paper Packaging was the first flexible packaging before the invention of plastic and aluminum foil. Currently, paper packaging is still widely used and is able to compete with other packaging such as plastic and metal because it is cheap, easy to obtain and widely used. The weakness of packaging paper for packaging food ingredients is that it is sensitive to water and is easily affected by environmental humidity.

Currently, the demand for domestic packaging paper continues to increase driven by the increasing level of consumer awareness, about sustainable packaging, together with strict regulations imposed by various environmental protection agencies, regarding the use of environmentally friendly packaging products, which is driving an increasing market for packaging based paper.

PT XYZ has been established since 1993 and is a local company engaged in the corrugated cardboard box (KKG) packaging industry in Indonesia, with an area of + 2.0 hectares in the western area of Purwakarta, West Java. and has a production capacity of +500 tons per month.

PT. XYZ produces using the Make to Order (MTO) system, where several production activities such as final assembly and component manufacturing wait until there is an incoming order from the customer. However, some activities, such as providing production capacity, are carried out on the basis of forecasting or predicting incoming orders, where prediction of incoming orders is an activity to estimate the size of incoming orders for certain goods in a certain period and marketing area. So predictive figures can be made for a monthly period. In the hierarchy of predictions, different models can be made.

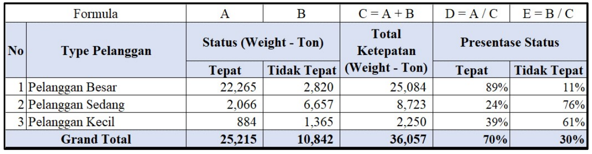
So far, the problem of fulfilling incoming orders every month is considered not optimal, as described in Table 1.1 which explains the status of the accuracy of the fulfillment of incoming orders which is divided into 2 (two) namely "Right" and "Incorrect", then percentage in weight (Tons) :

Table 1. Percentage of Accuracy of Distribution of Packaging Paper Requests

Status Ketepatan Pemenuhan	Jumlah Permintaan	Jumlah Weight (Ton)	Presentase Jlm Weight (Ton)
TEPAT	1,221	25,215	70%
TIDAK TEPAT	566	10,842	30%
Grand Total	**1,787**	**36,057**	**100%**

From Table 1 above, it can be seen that the percentage of accuracy in fulfilling incoming orders during 2019 was only around 70% and the inaccuracy in fulfilling paper requests was around 30%. Then in Table 2 explains the average percentage of fulfillment accuracy based on the type of customer (Large Customers, Medium Customers, and Small Customers) :

Table 2. Average Percentage of Demand Distribution Accuracy



Formula		A	B	C = A + B	D = A / C	E = B / C
No	**Type Pelanggan**	**Status (Weight - Ton)**		**Total Ketepatan (Weight - Ton)**	**Presentase Status**	
		Tepat	**Tidak Tepat**		**Tepat**	**Tidak Tepat**
1	Pelanggan Besar	22,265	2,820	25,084	89%	11%
2	Pelanggan Sedang	2,066	6,657	8,723	24%	76%
3	Pelanggan Kecil	884	1,365	2,250	39%	61%
Grand Total		**25,215**	**10,842**	**36,057**	**70%**	**30%**

In Table 2. above, it can be seen that the average percentage of the accuracy of fulfillment of incoming orders for large customers with an average accuracy of fulfillment of about 89% and inaccuracy of fulfillment of about 11%, medium customers with an average fulfillment of about 24% and inaccuracy of fulfillment of about 76% while for small customers the average accuracy of fulfillment is around 39% and the inaccuracy of fulfillment is around 61%.

This is because the ups and downs of incoming orders cannot be predicted properly. So that the production team has difficulty in planning the management of maximum production capacity to fulfill incoming orders. Figure 1 shows the trend of packaging paper demand during 2019 :

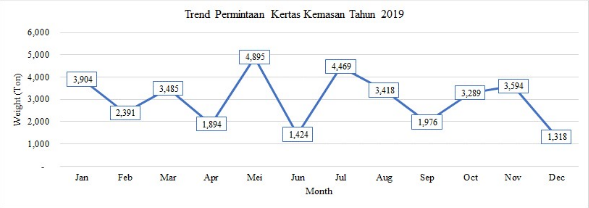


Figure 1. Incoming Order Trend in 2019 Tahun.

As a result of the problems caused above can give a bad predicate for the level of service provided by PT. XYZ on customers and also has the potential to reduce the company's profits.

Time series prediction methods such as Support Vector Machine (SVM) [2] [3], Recurrent Neural Network (RNN) [4] and LSTM [1] are proposed by many researchers to predict order.

RNN [4] has advantages in predicting sequential data, strong in each processing, RNN will store the internal state, namely St, which is given from one time step to the next time step. This is the *"Memory"* of the RNN, but has the disadvantage that the number of layers in the RNN itself can be very long, as long as the number of rows in the input, and this poses a problem in itself because the RNN often has to store dependencies from inputs that are located quite far apart which are commonly called *"Vanishing Gradient"*.

LSTM [1] can solve the RNN problem, namely Vanishing Gradient, because it has a different processing with ordinary RNN modules. Another difference is that additional signals are given from one time step to the next, namely the cell state and memory cell, represented by the symbol Ct. which is appropriate for the characteristics of the demand for packaging paper in this study.

In this study, LSTM will be applied to select the appropriate and optimal cell state and memory cells, so that the results of the prediction model for packaging paper demand are more accurate according to the company's needs. The expected result is that the demand prediction model for packaging paper using the Long Short-Term Memory (LSTM) method can help improve the management of packaging paper demand according to company needs and can also help improve PT. XYZ to customers with timely fulfillment and according to incoming orders.

Based on the background of the problem in this study, there is no good prediction of incoming orders. With the aim of being able to apply an incoming order prediction model with the Long Short-Term Memory (LSTM) Method.

II. RESEARCH METHOD

A. Prediction Theory

To solve problems in the future that cannot be ascertained, people always try to solve them with models of approaches that are in accordance with the actual behavior of the data, as well as in making predictions [13].

Predicting (forecasting) demand for products and services in the future and its parts is very important in planning and monitoring production [14]. A prediction has many meanings, so the prediction needs to be planned and scheduled so that it will take a period of time at least in the period of time needed to make a policy and determine several things that affect the policy.

Predictions are needed in addition to estimating what will happen in the future, decision makers also need to make plans.

B. Definition of Prediction

Prediction is an estimate of the expected level of demand for a product or several products in a certain period of time in the future. Therefore, Prediction is basically an estimate, but using certain methods Prediction can be more than just one estimate. It can be said that Prediction is a scientific estimate although there will be some errors due to the limitations of human abilities.

Before describing this Prediction method, it is first described about the definition of Prediction itself. Prediction is the activity of estimating the expected level of product demand for a product or several products in a certain period of time in the future [5].

According to Buffa: "Prediction or forecasting is defined as the use of statistical techniques in the form of a future picture based on the processing of historical figures" [6].

According to Makridakis: "Prediction is an integral part of management decision-making activities" [7].

Organizations always set goals and objectives, try to estimate environmental factors, and then choose actions that are expected to result in the achievement of these goals and objectives. The need for prediction increases in line with management's efforts to reduce its dependence on things that are uncertain. Prediction becomes more scientific in nature in the face of management environment. Because every organization is related to each other, good or bad forecasts can affect all parts of the organization [7].

C. Prediction Technique

Prediction techniques, in general, the time series method can be grouped into:

1. Averaging Method

Used for conditions where each data at different times has the same weight so that random fluctuations in data can be soaked with the average, usually used for short-term predictions [8]. The methods included in it, among others:

- Simple Average

$$F_{T+n} = \overline{X} = \sum_{i=n}^{T+(n-1)} \frac{X_i}{T}$$

Formula description:

X = F = Prediction results
T = Period
Xi = Demand in period t

- Simple Moving Average
  If stationary data is obtained, this method is good enough to predict the situation. Formula used :

$$F_{T+n} = \overline{X} = \frac{X_1 + X_2 + \dots\dots + X_n}{T}$$

Formula description:
X = F = Prediction results
T = Period
Xi = Demand in period t

- Double Moving Average
  Jika data tidak stasioner serta mengandung pole trend, maka dilakukan moving average terhadap hasil single moving average. Rumus yang digunakan :

$$S'_t = \frac{X_t + X_{t-1} + .. + X_{t-1}}{N}$$

2. Metode *Smoothing* (Pemulusan)

Used in conditions where the weight of the data in one period is different from the data in the previous period and forms an Exponential function which is commonly called Exponential smoothing [9]. The methods included in it, among others:

- *Single Exponential Smoothing*

This method greatly reduces the problem of data distortion because there is no need to store historical data anymore. The effect of the size of a is in the opposite direction to the effect of entering the number of observations. This method always follows any trend in the actual data because all it can do is set future forecasts with a percentage of the last error. To determine a close to optimal requires several trials. Formula used :

$$F_{t+1} = F_t + \alpha \times (X_t - F_t)$$

Where : Ft+1 = Prediction result t + 1
a = Smoothing constant
Xt = Demand in period t
Ft = Previous period

- *Double Exponential Smoothing* one parameter of *Browns*.

The rationale for Browns linear exponential smoothing is similar to that of a linear moving average, because both single and multiple smoothing values lag behind the actual data if there is an element of trend. The equation used in this method is as follows:

$$S'_t = aX_t + (1\text{-}a)S'_{t\text{-}1}$$

Formula description :
Xt = Demand in period t
S't = Smoothing value I period t
S"t = Smoothing value II period t
S't-1 = Previous first smoothing value (t-1)
S"t-1 = Previous second smoothing value (t-1)
a = Smoothing constant
at = Interception in period t
bt = Period trend value t
Ft+1 = Prediction Results for the period t+1
m = Number of forecasted future time periods

- *Regresi Linier*

Linear regression is used for prediction if the existing data set is linear, meaning that the relationship between the time variable and demand is in the form of a line (linear). The linear regression method is based on the calculation of the least square error, namely by calculating the smallest distance to a point in the data to draw a line. As for the linear regression prediction equation, three constants are used, namely a, b and Y [10]. With each formulation is as follows:

$$b = \frac{n\sum X_i Y_i - \sum X_i \sum Y_i}{n\sum X_i^{\,2} - \left(\sum X_i\right)^2}$$

Formula description :
y = Predicted Variables
a,b = Prediction Parameters
t = Independent variable

D. Research design

In this study, several steps will be taken to achieve the research objectives. These steps can be illustrated through the flow chart in Figure 2. This research starts from collecting data from raw data. The next stage is the determination of the network architecture design by determining the input and output patterns for training and testing purposes on an artificial neural network (ANN). This stage is then followed by the determination of the training algorithm.

Next is the training stage for the data that has been normalized and the architecture determined, the training is carried out first for the standard backpropagation algorithm, after that the training is carried out again by adding the learning rate and momentum coefficient to the weight update. The purpose of the training was to determine the value of the Root Mean Squared Error (RMSE) [11] and the Augmented Dickey–Fuller test (ADF) [12]. Carry out the testing phase of the test data, with the aim of knowing the level of validation of the results.

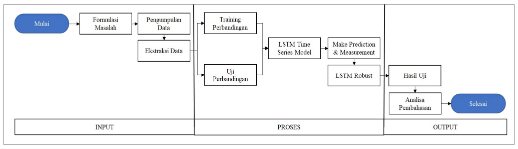


Figure 2. Research Design

The model experiment phase begins with model making and LSTM Network training. The training results are in the form of an LSTM model in the form of a CSV file. The model obtained is then used in the prediction process by loading the model file. The final stage of the experiment is the process of denormalizing the test data to get the predicted value and the evaluation value of the model's performance results. The benefit of the training and prediction processes being done separately is when running the prediction process if there is no new data. There is no need to model the training data again, but directly load it from the file. This will speed up the prediction process because without having to retrain the same data.

E. Data collection

The data used in this study is data on the realization of the demand for packaging paper at PT. XYZ from 2016 to 2019 in the form of secondary data that is quantitative. Consists of 8 attributes and 166.899 rows. The description of the realization of data on the fulfillment of packaging paper demand at PT. XYZ can be seen in table 3.

Table 3. Overview of Incoming Order Data

Order Date	Req. Ship. Date	Mills	Order Number	Cust. ID	Mat. ID	Prod. Group	SO Weight (KG)
3/18/2016	3/30/2019	NBL	2611004828	Cust1	MT001	LNR	19.80
3/18/2016	3/30/2019	NBL	2611004828	Cust3	MT002	LNR	19.20
3/18/2016	3/30/2019	NBL	2611004828	Cust1	MT003	LNR	19.60
5/15/2016	6/15/2019	NBL	2611004829	Cust4	MT002	LNR	24.00
5/15/2016	6/15/2019	NBL	2611004830	Cust2	MT003	LNR	23.80
1/13/2019	1/30/2017	NBL	2611006651	Cust39	MT096	LNR	30.11
1/6/2019	1/30/2017	NBL	2611006637	Cust19	MT053	MDM	20.47
1/6/2019	1/30/2017	NBL	2611006637	Cust19	MT021	MDM	13.10
…	…	…	…	…	…	…	…
12/3/2018	12/9/2020	NBL	2611003948	Cust94	MT259	LNR	0.11
12/4/2018	12/9/2020	NBL	2611003952	Cust76	MT424	WTP	1.00

F. Data processing

Before the implementation stage is implemented, the preprocessing stage is first carried out. The number of initial data that can be obtained from data collection is 1,335,192 records, but not all data is used and not all attributes are used because the data must go through the initial data processing stage or is called data preparation..

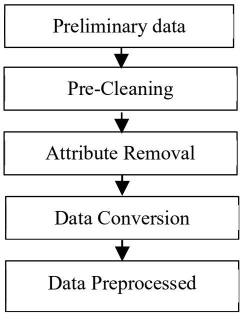


Figure 3. Data Processing

G. Proposed method

Long Short-Term Memory (LSTM) [1] was first mentioned in 1997 by Hochreiter and Schmidhuber. LSTM is also known as a neural network with an adaptable architecture, so its shape can be adjusted depending on the application. Below Figure 4 shows a model diagram for the LSTM method.

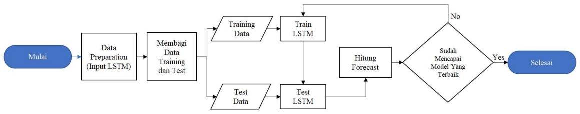

Mulai
Data Preparation (Input LSTM)
Membagi Data Training dan Test
Training Data
Test Data
Train LSTM
Test LSTM
Hitung Forecast
Sudah Mencapai Model Yang Terbaik
No
Yes
Selesai

Figure 4. LSTM Method Model Diagram

Long Short-Term Memory is a derivative of the RNN (Recurrent Neural Network) method. RNN is an iterative neural network which is specially designed to handle sequential data. However, RNN has a vanishing and exploding gradient problem, namely if there is a change in the range of values from one layer to another.

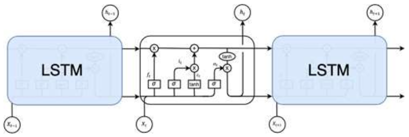


Figure 5. Long Short Term Memory (LSTM) Architecture

The hidden layer consists of memory cells, one memory cell has three gates, namely input gate, forget gate, and output gate. The input gate controls how much information should be stored in the cell state. This prevents the cell from storing unnecessary data. Forget gate functions to control the extent to which the value remains in the memory cell. Output Gate serves to decide how much content or value is in a memory cell, it is used to calculate output.

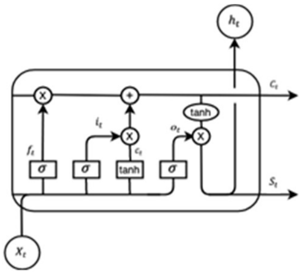


Figure 6. Shell Memory LSTM

Next on an architecture. The LSTM is built and designed to overcome the problem of gradient vanishing from RNNs when dealing with vanishing and exploding gradients. The LSTM architecture consists of an input layer, an output layer, and a hidden layer which is presented in Figure 5.

Figure 6 presents the contents of the hidden layer of LSTM namely memory cells. A memory cell in the LSTM stores a value or state (cell state), either for a long or short period of time. The explanation for the gates in one Long Short Term Memory (LSTM) memory cell is as follows:

1. Input Gate (i_t)
   The input gate takes the role of taking the previous output and the new input and passing them through the sigmoid layer. This gate returns a value of 0 or 1. The formula for i_t is:

$$i_t = \sigma(W_i S_t{-1} + W_i X_t)$$

Formula description :
$W i$ = Weight of Input Gate
$S t-1$ = Previous state at time t-1
Xt = Input on time t
σ = Sigmoid activation function

The input gate value is multiplied by the output of the candidate layer ($\tilde{C}$). Formula for ($\tilde{C}$) is :

$$\tilde{C} = tanh(W_c S_t{-1} + W_c X_t)$$
$$ct = (it * \tilde{C}_t + f_t * c_t{-1})$$

Formula description,
$\tilde{C}$ = Intermediate cell state.
$W c$ = Weight of cell state.
$S t-1$ = Previous state at time t– 1.
$X t$ = Input on time t.

The previous state is multiplied by the forget gate and then added to the new candidate function allowed by the output gate.

2. Forget Gate (f_t)
Forget gate is a sigmoid layer that takes the output at time t – 1 and input at time t and combines them and applies the sigmoid activation function. Since it is sigmoid, the output of this gate is 0 or 1. If $f_t = 0$ then the previous state will be forgotten, while if $f_t = 1$ the previous state has not changed. Formula of f_t is :

$$f_t \;=\; \sigma(W_f S_t{-}1 + W_f X_t)$$

Formula description,
$W f$ = Weight of forget gate.
$S t{-}1$ = Previous state at time t - 1.
$X t$ = Input on time t.
σ = Sigmoid activation function

This layer applies a hyperbolic tangent to the previous mix of input and output. Returns the candidate vector to be added to the state.

3. Output Gate (O_t)
The output gate controls how many states pass to the output and works in the same way as any other gate. And finally generate a new cell state (ht). Formula of $o\,t$ and ht is :

$$o\,t \;=\; \sigma(W\,o\,S\,t{-}1 + W\,o\,X\,t)$$
$$ht \;=\; o\,t \;*\, t\,a\,n\,h(c\,t\,)$$

Formula description,
$W\,o$ = Weight of output gate.
$S\,t{-}1$ = Previous state or current state t - 1.
$X\,t$ = Input on time t.
σ = Sigmoid activation function

The prediction accuracy is obtained from the data that has been trained and the key to the success of both is the number of hidden layers. There are two attributes used in this study, namely the date of sale and the value or value of daily income from orders entered by PT. XYZ. The LSTM Method Training Model diagram is shown in Figure 7 below 7:

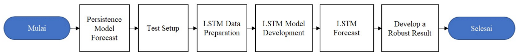


Figure 7. LSTM Method Training Model Diagram

H. Result Evaluation

For testing the performance of the model using the Root Mean Square Error (RMSE). Root Mean Square Error (RMSE) is an alternative method for evaluating forecasting techniques used to measure the accuracy of the forecast results of a model. The resulting value RMSE is the average value of the square of the number of errors in the prediction model. Root Mean Square Error (RMSE) is a technique that is easy to implement and has been frequently used in various studies related to RMSE forecasting which is expressed by the following formula :

Root Mean Square Error (RMSE)

$$\sqrt{\frac{1}{n}\sum\nolimits_{i}^{n}(y_i - y_i)^2}$$

Formula description:
$\tilde{y}\,i$ = Forecasting value
$y\,i$ = Actual value
n = Amount of data

I. Unit Root Test

Stationarity is one of the important prerequisites in time series data models. Stationary data is data that shows the mean, variance and auto variance (on the lag variation) remains the same at any time the data is formed or used, meaning that with stationary data the time series model can be said to be more stable. If the data used in the model is not stationary, then the data is reconsidered for its validity and stability.

One of the formal concepts used to determine the stationarity of data is through a unit root test. This test is a popular test, developed by David Dickey and Wayne Fuller (1979) as the Augmented Dickey-Fuller (ADF) Test [15]. If a time series data is not stationary at zero order, I(0), then the data stationarity can be searched through the next order so that the stationarity level is obtained on the nth order (first difference or I(1), or second difference or I(2).), etc. Several models can be selected to perform the ADF Test:

ΔYt = δYt-1 + ut (without intercept)
ΔYt = β + δYt-1 + ut (with intercept)
ΔYt = β1 + β2t + δYt-1 + ut (intercept with time trend)
Δ = first difference of the variables used
t = variabel trend

The hypothesis for this test is :
H0 : δ = 0 (there is a unit root, not stationary)
H1 : δ ≠ 0 (no unit root, stationary)

III. RESULT AND DISCUSSION

A. Data preparation

The data collected consists of 8 columns and 166.899 rows. The data is historical data on demand transactions with a time span from January 2016 to December 2019. The data is in the form of an excel file with 8 variables and 1,335,192 records, hereinafter referred to as paper-sales.

The variables in the transaction data include: Order Date, Req. Ships. Date, Factory, Order Number, Customer ID, Sales Person, Material ID, and SO Weight (MT).

Table 4. Display of Incoming Order Data

Order Date	Req. Ship. Date	Mills	Order Number	Cust. ID	Mat. ID	Prod. Group	SO Weight (KG)
3/18/2016	3/30/2019	NBL	2611004828	Cust1	MT001	LNR	19.80
3/18/2016	3/30/2019	NBL	2611004828	Cust3	MT002	LNR	19.20
3/18/2016	3/30/2019	NBL	2611004828	Cust1	MT003	LNR	19.60
5/15/2016	6/15/2019	NBL	2611004829	Cust4	MT002	LNR	24.00
5/15/2016	6/15/2019	NBL	2611004830	Cust2	MT003	LNR	23.80
1/13/2019	1/30/2017	NBL	2611006651	Cust39	MT096	LNR	30.11
1/6/2019	1/30/2017	NBL	2611006637	Cust19	MT053	MDM	20.47
1/6/2019	1/30/2017	NBL	2611006637	Cust19	MT021	MDM	13.10
…	…	…	…	…	…	…	…
12/3/2018	12/9/2020	NBL	2611003948	Cust94	MT259	LNR	0.11
12/4/2018	12/9/2020	NBL	2611003952	Cust76	MT424	WTP	1.00

Display This data set describes the number of monthly incoming orders for a period of 4 years, before testing the data shown in the figure:

```
Month
2016-01-01    36.8
2016-02-01    27.6
2016-03-01    29.3
2016-04-01    36.0
2016-05-01    37.4
Name: Sales, dtype: float64
```

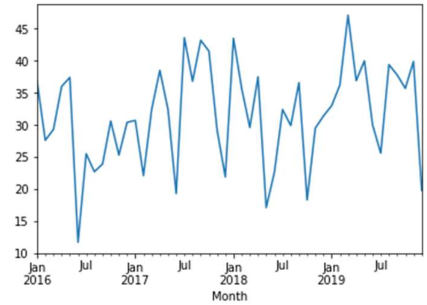


Figure 8. Display of Incoming Order Data Collection for 4 Years

B. Dataset Test Setting

The paper-sales dataset will be divided into 2 (two) parts, namely a training set and a test set. The first 2 (two) years of data will be taken for the training data set and the other 2 (two) years of data will be used for the test set.

The process of dividing the dataset or what is called the Train-Test Split is as follows :

```
Observations: 48
Training Observations: 36
Testing Observations: 12
```

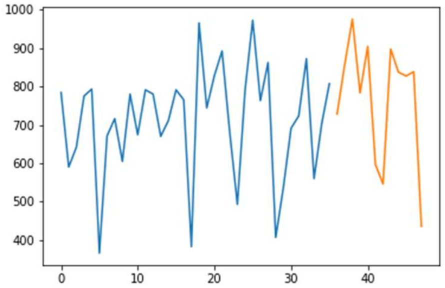


Figure 9. Display of Test Data Test

Furthermore, the model will be developed using the training dataset and will make predictions on the test dataset.

A rolling forecast scenario will be used, also called Walk-Fordward model validation, where each step of the test dataset will be executed individually, then the actual expected values from the test dataset will be fetched and made available to the forecast model at the next time step. This mimics a field scenario where research on sales of new packaging paper will be made available each month and used in the forecast for the following month.

From all estimates on the dataset will be collected error scores are calculated to summarize the skills of the model. The root mean squared error (RMSE) will be used because it can produce a score that is in the same unit as the estimated data, namely monthly sales of packaged paper.

C. Estimated Persistence Model

A good basic approximation for a time series with a linear upward trend is the persistence estimate. Persistence estimates where observations from the previous time step (t-1) are used to predict the observations in the current time step (t). We can implement this by taking the last observations from the training and history data accumulated with walk-forward validation and using them to predict the current time step.

The following is the persistence estimation model on the paper-sales-l dataset as follows::

```
RMSE: 9.054
```

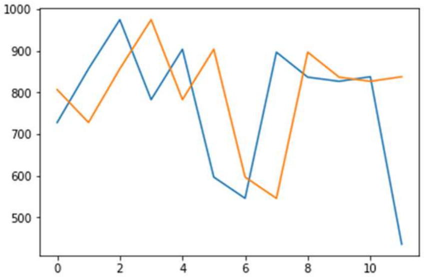


Figure 10. Display of the Persistence Model

In the persistence estimate model above, an RMSE value of around 198.7 monthly packaging paper sales is generated for the estimate on the test data set set.

D. Turning Datasets into Supervised Learning

The LSTM model in Keras assumes that the data will be divided into input (X) and output (y) components. For Time Series, it can be achieved by using the last time step observation (t-1) as input and the current time step observation (t) as output.

Then to combine these two series together to create a Data Frame that can be used in Supervised Learning. The series that is pushed down will have a new position at the top with no value. The NaN value (no numbers) will be used in this position which replaces this NaN value with a value of 0, which the LSTM model must learn as the start of the circuit. Here are the results:

```
       0      0
0    0.0   36.8
1   36.8   27.6
2   27.6   29.3
3   29.3   36.0
4   36.0   37.4
```

E. Converting Dataset to stationary data

The Packaging Paper Sales dataset is classified as non-stationary where there is a structure in the data that depends on time. Specifically, there is an increasing trend in the data. Stationary data are easier to model and are more likely to produce more skilled estimates perkiraan.

The standard way to clear a trend is to differentiate the data. That is, the previous research time step (t-1) is subtracted from the current study (t). This removes the trend and we are left with a series of differences, or changes in the study from one time step to the next. Here are the results:

```
Month
```

```
2016-01-01    36.8
2016-02-01    27.6
2016-03-01    29.3
2016-04-01    36.0
2016-05-01    37.4
Name: Sales, dtype: float64
0    -9.2
1     1.7
2     6.7
3     1.4
4   -25.7
dtype: float64
0    27.6
1    29.3
2    36.0
3    37.4
4    11.7
dtype: float64
```

From the results above, the first 5 rows of loaded data are printed, then the first 5 rows are from different series, then finally the first 5 rows with the difference operation being reversed. For the first study in the original dataset was removed from the data the inverse difference. Moreover, the last data set matches the first as expected.

F. Splitting the Dataset to Scale

Like other Neural Networks (NN), LSTM expects data to be within the scale of the activation function used by the network. The default activation function for the LSTM is the hyperbolic tangent (tanh), which returns a value between -1 and 1. This is the range of interest for time series data.

For a balanced experiment, the values of the scaling coefficients (min and max) must be calculated on the training dataset and applied to scale any test and forecast datasets. This avoids contaminating the experiment with knowledge from the test data set, which may give the model a small advantage.

Next convert the dataset to the range [-1, 1] using the MinMaxScaler class. Like other scikit-learn transform classes, it requires data to be provided in a matrix format with rows and columns. Therefore, we must reshape the NumPy array before performing the transformation.

```
Month
2016-01-01    36.8
2016-02-01    27.6
2016-03-01    29.3
2016-04-01    36.0
2016-05-01    37.4
Name: Sales, dtype: float64
0    0.418079
1   -0.101695
2   -0.005650
3    0.372881
4    0.451977
dtype: float64
0    36.8
1    27.6
2    29.3
3    36.0
4    37.4
dtype: float64
```

G. Estimate using the Long-Short Term Memory (LSTM) Model

Once the LSTM model matches the training data, the model can be used to make estimates. In this study it has some flexibility which can decide to adjust the model once on all training data, then predict each new time step one by one from the test data (fixed approach), or can adjust the model or update the model each time step from the test data as new observations from test data available (dynamic approach).

To make an estimate, we can call the predict() function on the model. This takes the NumPy 3D array input as an argument. In that case, it would be a layer with one value, research on the previous time step.

To run this model, you can call a function called forecast(). With model fit, the batch size used when adjusting the model (for example 1), and a row of test data, the function will separate the input data from the test row, reshape it, and return the prediction as a single floating point value.

During training, the internal state is reset after each epoch. When estimating in this study did not reset the internal state among estimates. In fact, the model builds state as we estimate each time step in the test data set set.

The scaling and reverse-scaling behavior has been moved to the scale() and invert_scale() functions for simplicity. The test data is scaled using a fit of scaler on the training data, as needed to ensure the min/max values of the test data do not affect the model. The sequence of data transformations is adjusted for convenience, first creating stationary data, then supervised learning problems, then scaling. Distinctions are made on the entire data set before being broken down into training and test sets for convenience. In this case, it is easy to collect observations during validation going forward and differentiate them as we proceed where it is not decided not to read them further. Below is the result :

```
Month=1, Predicted=25.133321, Expected=33.000000
Month=2, Predicted=26.136585, Expected=36.200000
Month=3, Predicted=28.210680, Expected=47.100000
Month=4, Predicted=38.214326, Expected=36.900000
Month=5, Predicted=32.778191, Expected=40.000000
Month=6, Predicted=34.051595, Expected=30.000000
Month=7, Predicted=33.395497, Expected=25.600000
Month=8, Predicted=33.054117, Expected=39.400000
Month=9, Predicted=34.184841, Expected=37.800000
Month=10, Predicted=33.495201, Expected=35.700000
Month=11, Predicted=33.384978, Expected=39.900000
Month=12, Predicted=34.004823, Expected=19.800000
Test RMSE: 8.914
```

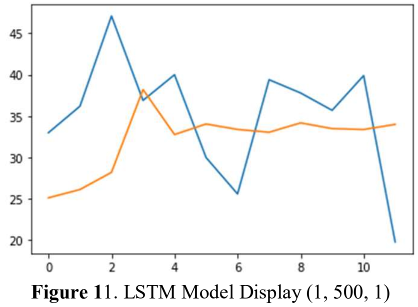


Figure 11. LSTM Model Display (1, 500, 1)

```
Month=1, Predicted=31.046228, Expected=33.000000
Month=2, Predicted=32.255115, Expected=36.200000
Month=3, Predicted=33.345093, Expected=47.100000
Month=4, Predicted=39.694085, Expected=36.900000
Month=5, Predicted=38.614510, Expected=40.000000
Month=6, Predicted=38.535921, Expected=30.000000
Month=7, Predicted=39.055871, Expected=25.600000
Month=8, Predicted=36.120883, Expected=39.400000
Month=9, Predicted=35.680632, Expected=37.800000
Month=10, Predicted=39.032655, Expected=35.700000
Month=11, Predicted=39.866720, Expected=39.900000
Month=12, Predicted=40.316062, Expected=19.800000
Test RMSE: 8.754
```

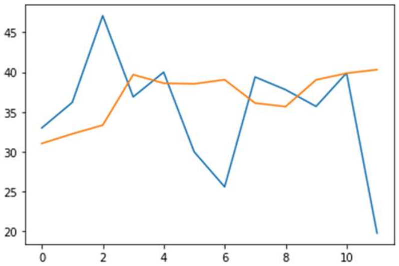


Figure 12. LSTM Model Display (1, 1500, 1)

```
Month=1, Predicted=26.131302, Expected=33.000000
Month=2, Predicted=27.513795, Expected=36.200000
Month=3, Predicted=30.229067, Expected=47.100000
Month=4, Predicted=40.482170, Expected=36.900000
Month=5, Predicted=38.248608, Expected=40.000000
Month=6, Predicted=35.598107, Expected=30.000000
Month=7, Predicted=33.753025, Expected=25.600000
Month=8, Predicted=30.770659, Expected=39.400000
Month=9, Predicted=33.095257, Expected=37.800000
Month=10, Predicted=33.409154, Expected=35.700000
Month=11, Predicted=32.137640, Expected=39.900000
Month=12, Predicted=33.928015, Expected=19.800000
Test RMSE: 8.576
```

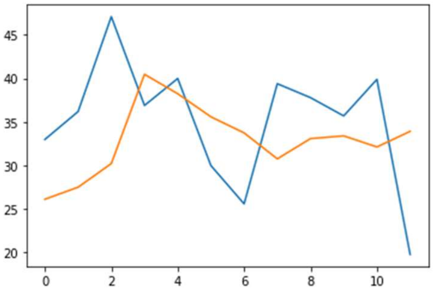


Figure 13. LSTM Model Display (1, 5000, 1)

H. Developing Robust Results

The problem with Neural Network (NN) is that this model gives different results to the initial conditions. One approach might be to improve the Random Number Seed used by Keras to ensure the results are reproducible. Another approach would be to control for random initial conditions using different experimental settings.

Next, the walk-forward model and validation will be placed in a loop with a fixed number of repetitions. Each run iteration of the RMSE can be recorded. We can then summarize the distribution of the RMSE scores. Below is the result:

```
1) Test RMSE: 8.735
2) Test RMSE: 9.242
3) Test RMSE: 8.620
4) Test RMSE: 8.716
5) Test RMSE: 8.544
6) Test RMSE: 8.818
7) Test RMSE: 8.509
8) Test RMSE: 8.628
9) Test RMSE: 8.590
10) Test RMSE: 8.544
...
29) Test RMSE: 8.659
30) Test RMSE: 10.027

rmse
count  30.000000
mean    8.767582
std     0.287924
min     8.508875
25%     8.621584
50%     8.711215
75%     8.782992
max    10.026662
```

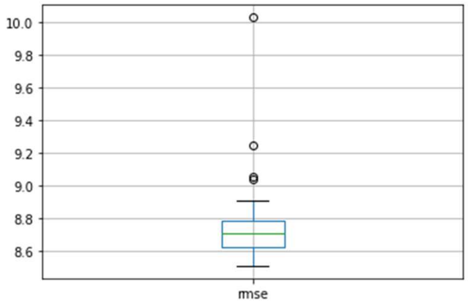


Figure 14. Robust Result Display (1, 5000, 1)

From the results above, it can be seen that the average value and standard deviation of the RMSE are 8.767582 and 0.287924 monthly incoming orders. A box and whisker plot is created from the distribution shown below. It captures the middle of the data as well as the extent and outliers. Then for measurement.

I. Augmented Dickey-Fuller test (ADF)

Statistical tests make strong assumptions about the dataset used. ADF to inform the extent to which the null hypothesis can be rejected or failed to be rejected. The results must be interpreted in order for certain problems to be meaningful.

The null hypothesis of this test is that the time series can be represented by a unit root, which is not stationary (has some time-dependent structure). The alternative hypothesis (rejecting the null hypothesis) is that the time series does not move.

- Hypothesis Zero (H0): If it fails to be rejected, this indicates that the time series has a unit root, meaning it is not stationary. It has some time-dependent structure.
- Alternative Hypothesis (H1): The null hypothesis is rejected; it shows the time series has no unit root, which means it doesn't move. It has no time-dependent structure.

The result uses the p-value of the test. A p-value below the threshold (such as 5% or 1%) indicates we rejected the null hypothesis (stationary), otherwise, a p-value above the threshold indicates we failed to reject the null hypothesis (non-stationary)..

- p-value> 0.05: Failed to reject the null hypothesis (H0), the data has a unit root and is non-stationary.
- p-value <= 0.05: Reject the null hypothesis (H0), the data has no unit root and is stationary.

Below is an example of calculating the Augmented Dickey-Fuller test on the Incoming Orders dataset for each product :

Table 5. ADF Recap for MDM, LNR and WTP Products

Measurement	MDM	LNR	WTP
ADF Statistic	-6.137597	-6.753697	-4.872927
P-value	0.000000	0.000000	0.000000
1%	-3.578	-3.578	-3.578
5%	-2.925	-2.925	-2.925
10%	-2.601	-2.601	-2.601

J. Comparison of LSTM with DES and Linear Regression

Before conducting research using the LSTM method, a comparison has been made with the Double Exponential Smoothing (DES) method [9] and Linear Regression [10] using the Minitab software. This is done to take the lowest error, of the three methods. Below table 6 shows the comparison:

Table 6. Comparison of LSTM, DES, Linear Regression Errors

Measurement	LSTM	DES	Regresi Linier
MSE	8.576	10.689	9.3279

From table 6 it can be seen that the lowest error using the Long Short-Term Memory (LSTM) method is 8.576.

K. *Fit* comparison of LSTM Models

In this study, the network parameters will not be adjusted, instead will use a comparison of the 3 configurations below, the RMSE and ADF Test will be measured:

Table 7. Comparison results of LSTM.

Measurement *fit* Model LSTM	*Batch:* 1 *Epochs:* 500 *Neuron:* 1	*Batch:* 1 *Epochs:* 1500 *Neuron:* 1	*Batch:* 1 *Epochs:* 5000 *Neuron:* 1
RMSE	8.914	8.754	8.576
ADF Test	ADF Statistic: -6.137597 p-value: 0.000000 1%: -3.578 5%: -2.925 10%: -2.601		

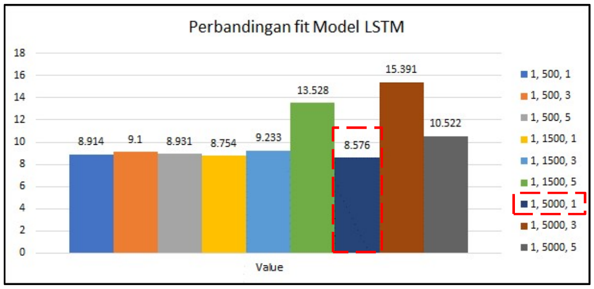


Figure 15. Comparison graph of LSTM . Model fit

From the results of the comparison of the fit of the LSTM model above, it can be seen that the parameter size of Batch: 1 Epochs: 5000 Neuron: 1 shows RMSE 8,576 lower than the size of the other LSTM model fit parameters, and the ADF Statistic value is -6.137597 The more negative this statistic, the more likely we are to rejecting the null hypothesis means that the dataset is stationary and the p-value 0.000000 means that the data does not have a unit root and is stationary.

IV. CONCLUSION

Based on the discussion that has been carried out, the conclusions are :

1. From the analysis of the trials that have been carried out, it can be concluded that the LSTM model is able to produce time series data predictions in this case, namely incoming orders with a small and accurate error rate. The average value and standard deviation of RMSE for MDM products are 8.767582 and 0.287924, RMSE for LNR products are 10.623984 and 0.466621, RMSE for WTP products are 1.636849 and 0.361515 monthly incoming orders.
2. The test results show that the number of 1 LSTM batch, epoch = 1500, and LSTM unit = 1 is the most optimal model parameter. The composition of the data sharing affects the prediction accuracy results. The best composition is obtained at 80% train data and 20% test data.
3. Predictions for the next 12 months are likely to remain stable. The future prediction data generated is between the range of initial orders for MDM products from 25.986923 to the last data, which is 34.050922, for LNR products from 39.865912 to the last data, which is 27.63460, for WTP products from 3.758679 until the last data is 2.678200.

REFERENCES

[1] Hochreiter, S., & Schmidhuber, J. (1997). LSTM 1997. *Neural Computation*.

[2] Ou, Y., Qian, H., & Xu, Y. (2009). Support vector machine based approach for abstracting human control strategy in controlling dynamically stable robots. Journal of Intelligent and Robotic Systems: Theory and Applications. https://doi.org/10.1007/s10846-008-9292-8

[3] Qian, M., Hongquan, W., Yongsheng, W., & Yan, Z. (2008). SVM based prediction of Spontaneous Combustion in Coal Seam. Proceedings of the 2008 International Symposium on Computational Intelligence and Design, ISCID 2008. https://doi.org/10.1109/iscid.2008.193

[4] Tseng, Y. C., Chen, C. C., Lee, C., & Huang, Y. K. (2007). Incremental in-network RNN search in wireless sensor networks. Proceedings of the International Conference on Parallel Processing Workshops. https://doi.org/10.1109/ICPPW.2007.47

[5] Biegel, J. E. (1961). Statistics in Forecasting. Management International, 162-181.

[6] Buffa, Elwood S., dan Sarin, R. K. (1996). Manajemen Operasi dan Produksi Modern. Edisi 8. Jakarta: Binarupa Aksara.

[7] Wheelwright, S., Makridakis, S., Makridakis, S., Wheelwright, S., Gross, C. W., Peterson, R. T., … O'Neill, W. J. (1978). Forecasting Methods for Management. Journal of Marketing Research. https://doi.org/10.2307/3150640

[8] Mitropolsky, I. A. (1967). Averaging method in non-linear mechanics. International Journal of Non-Linear Mechanics, 2(1), 69-96.

[9] Kitagawa, G. (1991). A nonlinear smoothing method for time series analysis. Statistica Sinica, 371-388.

[10] Andrews, D. F. (1974). A robust method for multiple linear regression. Technometrics, 16(4), 523-531.

[11] Willmott, C. J. (1981). On the validation of models. Physical geography, 2(2), 184-194.

[12] Cheung, Y. W., & Lai, K. S. (1995). Lag order and critical values of the augmented Dickey–Fuller test. Journal of Business & Economic Statistics, 13(3), 277-280.

[13] Rescher, N. (1998). Predicting the future: An introduction to the theory of forecasting. SUNY press.

[14] Syntetos, A. A., Babai, Z., Boylan, J. E., Kolassa, S., & Nikolopoulos, K. (2016). Supply chain forecasting: Theory, practice, their gap and the future. European Journal of Operational Research, 252(1), 1-26.

[15] Dickey, David A dan Wayne A. Fuller, (1979), Distribusi of Estimators for Autoregressive Time Series With a Unit Root, Journal of the American Statistical Association, Vol. 74, No. 366