KOMPUTASI: JURNAL ILMIAH ILMU KOMPUTER DAN MATEMATIKA
VOL. 23 (2) (2026), 1-11 p-ISSN: 693-7554, e-ISSN:2654-3990

Prediction And Analysis of Factors Affecting Marketplace Sales Using A Bidirectional LSTM Model For Inventory Estimation

Indah Cahyani[1], Tjut Awaliyah Zuraiyah*[2], Aries Maesya[3], Fitri Mintarsih[4]

[1,2,3]Department of Computer Science, Faculty of Mathematics and Natural Science, Pakuan University, Bogor, West Java, 16143, Indonesia

[4] Departement of Informatics Engineering, Faculty of Science and Technology, State Islamic University Syarif Hidayatullah, Jakarta

Abstract

Forecasting weekly sales at the product level can help marketplace sellers avoid both excess inventory and stock shortages. The purpose of this study is to investigate the sales forecasting of the Shopee store antianshop by applying the Bidirectional Long Short-Term Memory (BiLSTM) method in conjunction with correlation-based feature selection to predict the inventory. The raw dataset contained 5,426 weekly product records from Shopee Seller Centre covering 22 May 2023 to 31 May 2025; 5,312 records remained after preprocessing. For each product, the data were ordered by week and converted into ten-week input sequences. Pearson correlation showed that Add to Cart (r = 0.5835$) and Enter Cart (r = 0.5579$) were the strongest retained predictors of weekly Units Sold. BiLSTM was then compared with a unidirectional LSTM under the same experimental settings for ten products. LSTM recorded slightly lower average errors, with MAE of 1.7462 units, RMSE of 2.3977 units, and non-zero MAPE of 69.57%, while BiLSTM produced 1.8505 units, 2.4611 units, and 71.18%, respectively. The results indicate that BiLSTM was competitive but did not outperform the simpler LSTM model consistently. We combined the two models in a Streamlit dashboard that shows product forecasts and weekly inventory guidance. Generalizability of the findings should be done cautiously since the analysis used one store, relatively short product histories, and no systematic hyperparameter search.

Keywords: *BiLSTM; demand forecasting; inventory management; marketplace; Pearson correlation*

1. Introduction

Reliable demand estimates are becoming more important for e-commerce sellers who have to decide how much stock to hold for individual products. Weekly sales for online stores are often sporadic. Overestimation of demand can result in capital tied up in slow moving inventory. Underestimation of demand can result in stockouts and lost sales. Studies of intermittent online sales and retail demand also suggest that product-level patterns are heterogeneous and a single forecasting assumption might not fit all items [21], [22]. Marketplace data brings additional complexity because customer activity can change quickly between visits, page views, search clicks, likes, and cart activity.

Deep learning is frequently used to tackle multivariate time-series forecasting because these models can learn nonlinear relationships between successive observations. BiLSTM is an extension of the standard LSTM, which reads the same historical input in two directions and thus can form separate representations of the observed sequence [6], [23]. In this work, both directions are limited to the ten historical weeks provided as input; the model never receives the target week that it is asked to predict. Whether the reverse pass adds useful information is tested directly by comparing BiLSTM with a conventional one-directional LSTM.

A range of methods has been reported for sales and financial forecasting. Syakir et al. [7] applied linear regression to Shopee sales, whereas Afrianto et al. [8], Puteri [9], Puteri et al. [10], and Putra and Nurmawati

*Corresponding author. *E-mail address: tjut.awaliyah@unpak.ac.id*
Received: 20 February 2026, Accepted: 29 July 2026 and available online 31 July 2026
DOI: https://doi.org/10.33751/komputasi.v23i2.96

[11] examined BiLSTM in financial time-series problems. Karin et al. [12] used a related architecture for air-quality prediction. Recent international work on online and retail demand further highlights the value of product-specific modeling, engineered predictors, and comparisons among competing algorithms [21]-[22]-[24]-[25]. Taken together, these studies suggest that sequence models can be useful, but their performance still depends on the demand pattern, the selected inputs, and the evaluation procedure.

Three issues remain particularly relevant for small marketplace sellers. Much of the earlier literature addresses price movements, aggregate demand, or fields outside e-commerce rather than weekly sales for individual marketplace products. Also marketplace behavior variables are not always tested for their relationship to actual Units Sold. Forecasts are also frequently given as model outputs and are not converted into simple guidance for routine stock review. This work tackles these problems by integrating product-level BiLSTM forecasting, Pearson-correlation screening, and a Streamlit dashboard with real data from one Shopee seller. The contribution is not a new BiLSTM architecture but the practical integration of these components.

The aim of this study is to develop and assess an antianshop product-level BiLSTM forecasting system, determine behavioral variables associated with weekly Units Sold, and display the forecasts in a way that may aid inventory review. In particular, the study makes the following contributions:

1) a forecasting workflow that models individual products based on 2 years of chronologically sorted Shopee Seller Centre records;
2) a correlation-based screening step that relates engagement and purchase-intention indicators to weekly Units Sold;
3) a like-for-like comparison of BiLSTM and LSTM with the same predictors, ten-week windows, chronological validation split, model size and training configuration; and
4) an interactive dashboard to translate model outputs into actionable weekly stock prioritization information.

Section 2 introduces the dataset, preprocessing steps, predictor selection, model design, evaluation procedure and deployment. Section 3 discusses the experimental results and their operational interpretation. Section 4 concludes the paper with the main conclusions, study limitations and directions for further work.

2. Methods

The analysis was performed with a data-mining workflow, written in Python with TensorFlow, Keras and Scikit-learn. CRISP-DM was modified to seven working stages: Business Understanding, Data Understanding, Data Preprocessing, Correlation-Based Feature Selection, Modeling, Model Evaluation and Deployment [14]. The stages were not strictly sequential, because evaluation results could feed back into earlier decisions about preprocessing or modeling. The complete sequence is depicted in Figure 1.

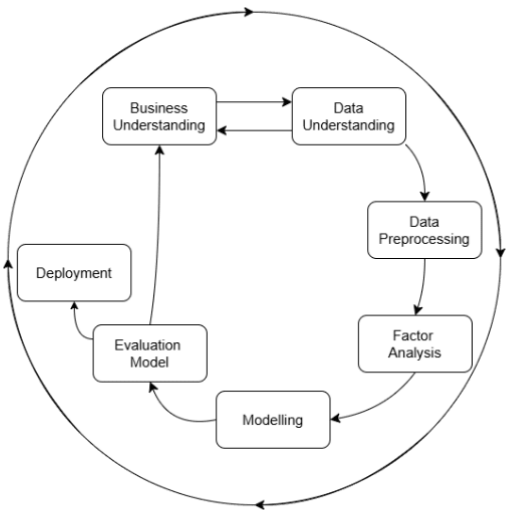


Figure 1. Research Flow Diagram

2.1. Data Collection and Understanding

Data collected directly from antianshop Shopee Seller Centre account. They span the period from 22 May 2023 to 31 May 2025 and contain initially 5,426 rows and 32 columns. A row represents one product record for a weekly reporting interval. Available fields include product identifiers, price, Units Sold, visits, page views, likes, cart activity, buyer information, and conversion measures. Table 1 gives an anonymized example of the original structure.

Table 1. Sample of the Historical Sales Dataset

No.	Product Code	...	Price	Units Sold
1	9392944615	...	18900	0
2	10060817529	...	15200	1
3	10060817529	...	28900	1
4	10445853847	...	28900	1
...	...	...	...	...
5423	29629937135	...	26900	0
5424	29713986816	...	16990	0
5425	29729926032	...	26890	0
5426	29929922540	...	26800	0

2.2. Data Preprocessing

Before modeling, the dataset was cleaned and reorganized through the following preprocessing steps:

a. Data Cleaning: Dash symbols used as invalid entries were converted to missing values. Where possible, a missing value was filled from another record with the same Product Code and the same weekly period. No value was taken from a later target week. The retained predictors were checked again for missing entries before sequences were created.

b. Data Filtering: Records for deleted products and product variations whose status was not Normal were excluded. This step left 5,312 rows for further analysis.

c. Feature Management: Fields used only for administration, including Variation Code and Parent SKU, were removed. After this reduction, 16 columns remained, and the variables used in the manuscript were assigned consistent English names.

d. Feature Engineering: Time translated to datetime format WeekStart and WeekEnd represented the weekly interval. Observations were then ordered by Product Code and date.

e. Data Normalization: The numerical variables are normalized in the range of [0,1] using Min-Max normalization as given in Equation (1). A common scale enables stable gradient-based training and prevents variables with large numerical values from dominating the optimization process.

$$x^1 = \frac{x - \min(x)}{\max(x) - \min(x)} \quad (1)$$

x is the raw observation, x' is the normalized observation and min(x) and max(x) are the minimum and maximum for the corresponding variable in Equation (1). For the controlled comparison each product was assigned its own scalers. Scalers were fitted only on the training period of that product, and then applied to validation observations without refitting. In this method information about the validation range did not get into the training transformation.

2.2.1 Time-Series Sequence Construction and Chronological Split

Weekly records were first arranged from the earliest to the latest date for each product. If d is the number of retained predictors, ten consecutive weeks from t-9 to t form an input array of size (10, d), and Units Sold at t+1 serves as the target. The next window is shifted by one week to generate the next input-target pair. Ten weeks is a fair compromise: it retains current history, but yet provides enough sequences for products with short records. Not selected by broad tuning. One sequence and target could be built from at least 11 weekly observations. The resulting samples were split in time order, the first 80% were used for training and the latest 20% were reserved for validation. Validation targets were never part of the training targets, but rolling windows could overlap with previous historic observations. The BiLSTM-LSTM comparison was made on the ten products with the highest total sales.

2.3. Correlation-Based Feature Selection

A Pearson correlation was generated prior to training the model to screen numeric variables linked with Units Sold. We call this step correlation-based feature selection here and not factor analysis, because it is based on the pairwise correlations of the predictors. Variables with |r| >= 0.40 were kept, based on the interpretation that this magnitude represents at least a moderate linear relationship [17]. Also the full matrix was examined for strong correlations between predictors, which may indicate redundant information. Pearson correlation simply describes linear connection and cannot be used to infer causality. The variables kept should thus be interpreted as predictors and not as causal factors. The coefficient utilized in the analysis is given in equation (2).

$$r = \frac{n\sum xy - \sum x \sum y}{\sqrt{(n\sum x^2 - (\sum x)^2)(n\sum y^2 - ((\sum y)^2)}} \tag{2}$$

Where r is the pearson coefficient, n is the number of paired observations, x is a predictor and y is the target variable Units Sold [18].

2.4. Modeling (BiLSTM and LSTM Baseline)

The main forecasting architecture was BiLSTM. One LSTM reads from oldest to newest observation, whereas a second reads the same observed weeks in reverse order. Dense layers concatenate concealed outputs. Since both branches use only the historical window, neither can access the unseen target week. A standard LSTM with the same recurrent and dense layers served as the baseline and differed only in processing direction. This setup isolates the effect of bidirectional processing [9], [23]. Figure 2 depicts the architecture.

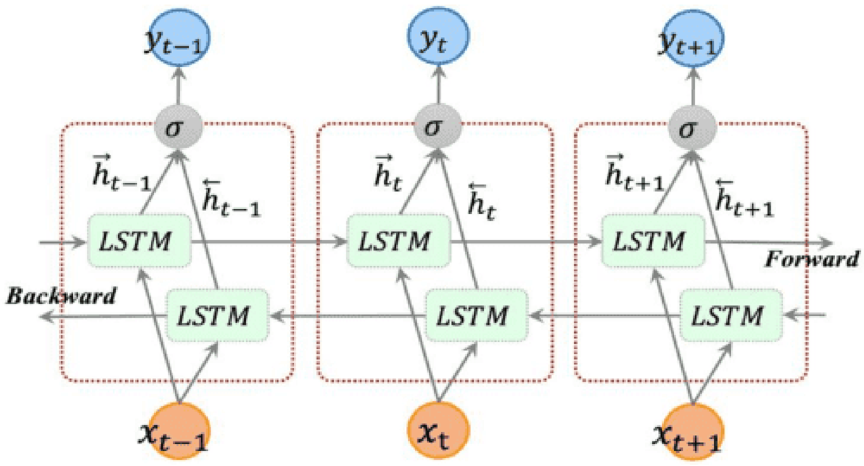


Figure 2. BiLSTM Architecture

Equations (3), (4), and (5) describe the forward LSTM, the backward LSTM, and their combined output, respectively.$\vec{h}_t\overleftarrow{h}_t h_t$

$$\vec{h}_t = LSTM(x_t, h_{t-1}) \tag{3}$$

$$\overleftarrow{h}_t = LSTM(x_t, h_{t+1}) \tag{4}$$

$$h_t = [\vec{h}_t, \overleftarrow{h}_t] \tag{5}$$

For every product included in the experiment, one BiLSTM and one LSTM model were fitted. Table 2 lists the architectures and the hyperparameters shared by both models.

Per-Product Modeling Rationale

Separate models were fitted because sales frequency, popularity, and engagement vary across products. A global model would provide more observations but could blur item-specific patterns. The per-product design preserves local behavior at the cost of smaller training sets [22].

Products required at least 11 ordered weekly records, which is enough to form one ten-week input and its target. This minimum does not ensure stable generalization, particularly for sparse series or products with repeated zero-sales weeks.

Table 2. Model and Baseline Hyperparameter Configuration

Model	Architecture	Training Setup	Optimization
BiLSTM	BiLSTM(64) + Dense(32, ReLU) + Dense(1)	Window = 10; batch = 32; max. 1,000 epochs; Early Stopping	Adam, learning rate = 0.01; MSE loss
LSTM	LSTM(64) + Dense(32, ReLU) + Dense(1)	Window = 10; batch = 32; max. 1,000 epochs; Early Stopping	Adam, learning rate = 0.01; MSE loss

The settings used in the controlled experiment are reported in Table 2. Each model contained 64 recurrent units, followed by a Dense layer with 32 ReLU units and a single linear output neuron. Training used the Adam optimizer, a learning rate of 0.01, and Mean Squared Error loss. The input window was fixed at ten weeks, the batch size at 32, and the maximum number of epochs at 1,000. Early Stopping monitored validation loss and restored the best weights. Apart from the bidirectional wrapper used only in BiLSTM, the two configurations were the same. These values were fixed implementation choices and were not obtained from a systematic hyperparameter search.

2.5. Model Evaluation

Performance was measured on the most recent validation sequences, which had been held out chronologically. Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) were computed using Equations (6) and (7). MAE gives the average absolute difference between predicted and observed sales, whereas RMSE penalizes larger deviations more heavily. Before either metric was calculated, predictions and targets were returned to the original Units Sold scale. Smaller values indicate more accurate forecasts [24], [25].

$$MAE = \frac{1}{n}\sum_{t=1}^{n} |f_t - y_t| \tag{6}$$

$$RMSE = \sqrt{\frac{\sum_{t=1}^{n} (Y'_t - Y_t)^2}{n}} \tag{7}$$

In both formulas y is the observed value and y-hat is the prediction from the model. Also reported was the Mean Absolute Percentage Error (MAPE) but only for the validation weeks where the actual sales were greater than zero. This avoids division by zero. Because a one-unit error might provide a very big percentage when the actual value is modest, the study relies mostly on MAE and RMSE in original units, treating non-zero MAPE as extra evidence.

2.6. Deployment

The models were then integrated to an interactive dashboard developed with Streamlit. The framework facilitates data-science web applications without a separate conventional front-end implementation [20]. Store owner can login to the dashboard and check the correlation results, the model performance, the sales forecast and the recommended priorities for the weekly stock management [20].

3. Result and Discussion

The developed application is a combination of predictor screening, BiLSTM forecasts at the product level, validation plots, multi-week outlooks and weekly stock recommendations. It's designed to facilitate decision making, not necessarily to create purchase orders in an automated way. The user still needs to consider available capital, current inventory, promotional plans, supplier lead time, and other business conditions.

In the interface, the five modules are offered, including data overview, correlation analysis, model prediction, multiweek forecasting, and inventory plan. Figure 3. English language dashboard from the revised article.

A seller can ask for a forecast up to 12 weeks ahead. The strategy module then ranks products by predicted demand and attaches simple guidance to items with relatively high or low forecasts for the following week..

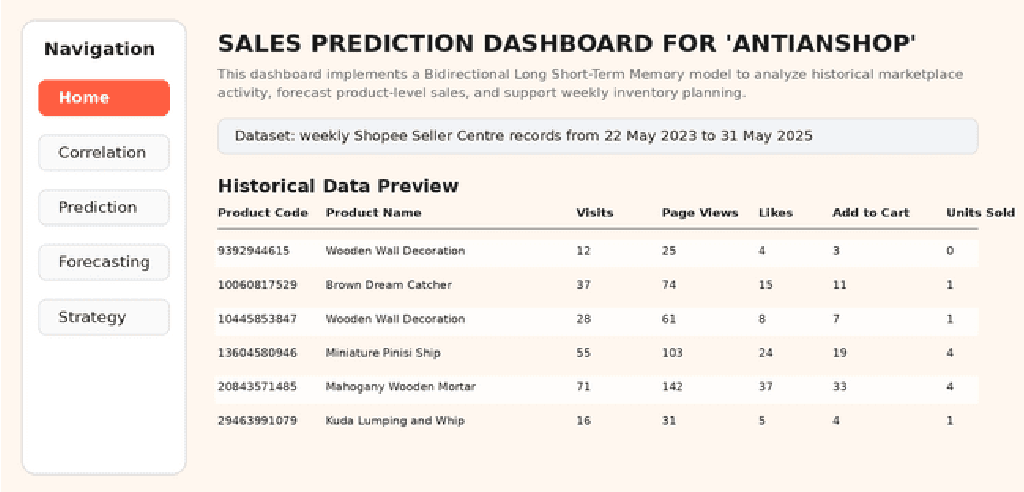


Figure 3. Website Interface

3.1. Data Preprocessing

The data cleaning, filtering, field selection, construction and normalization of time variables reduced the dataset from 5,426 data rows and 32 fields to 5,312 rows of data and 17 final fields.Figure 4 shows how invalid dash entries were replaced, and Table 3 records the dataset size after each preprocessing step.

Administrative or inactive data are not regarded as current demand by deleting deleted products and non-normal variations. 5,312 doesn't mean number of sequences used for training it means number of rows. Counts of sequences are fewer and varies by product.

```
df_dataawal = df_dataawal.replace('-', pd.NA)
```

Figure 4. Removal of Invalid Values

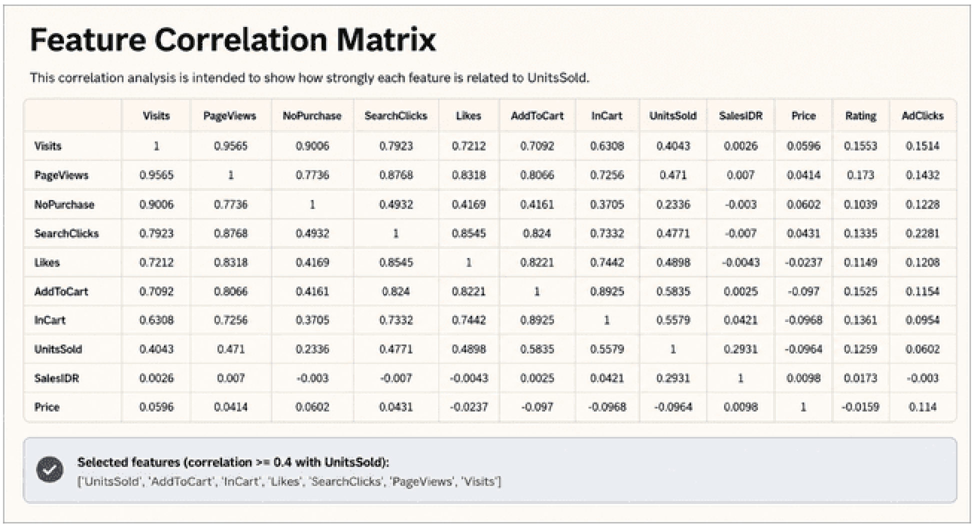
Table 3 traces the change in dataset dimensions from the raw file to the final normalized dataset.

Table 3. Data Preprocessing Steps

Step	Description	Result
Raw Data	Original dataset from Shopee Seller Centre.	5426 rows, 32 columns
Data Cleaning	Replaced invalid ("-") values with NA.	5379 rows, 28 columns
Data Filtering	Removed deleted or non-normal product variations.	5312 rows, 20 columns
Feature Selection	Dropped irrelevant attributes.	5312 rows, 16 columns
Feature Engineering for Time	Converted time column, created WeekStart & WeekEnd.	5312 rows, 17 columns
Normalization	Scaled all numerical variables into range [0,1].	5312 rows, 17 columns (final dataset)

3.2. Results of Correlation-Based Feature Selection

The largest retained correlations with Units Sold were obtained for Add to Cart ($r = 0.5835$) and Enter Cart ($r = 0.5579$). They were followed by Likes ($r = 0.4898$), Search Clicks ($r = 0.4771$), Page Views ($r = 0.4710$), and Visits ($r = 0.4043$). Cart-related measures sit closer to an actual purchase than general exposure measures, which may explain why they were more informative for short-term prediction in this store. These coefficients indicate association, not cause and effect. Visits and Page Views were also strongly related to each other ($r = 0.9665$), so the two variables may contain overlapping information. Keeping both may preserve slightly different platform measurements, but it may also increase model complexity; an ablation or regularization study would be needed to test whether both are necessary. Price ($r = -0.0984$) and Rating ($r = 0.1259$) fell below the threshold and were not used as predictors. Figure 5 shows the correlation matrix with English labels.



Feature Correlation Matrix

This correlation analysis is intended to show how strongly each feature is related to UnitsSold.

	Visits	PageViews	NoPurchase	SearchClicks	Likes	AddToCart	InCart	UnitsSold	SalesIDR	Price	Rating	AdClicks
Visits	1	0.9565	0.9006	0.7923	0.7212	0.7092	0.6308	0.4043	0.0026	0.0596	0.1553	0.1514
PageViews	0.9565	1	0.7736	0.8768	0.8318	0.8066	0.7256	0.471	0.007	0.0414	0.173	0.1432
NoPurchase	0.9006	0.7736	1	0.4932	0.4169	0.4161	0.3705	0.2336	-0.003	0.0602	0.1039	0.1228
SearchClicks	0.7923	0.8768	0.4932	1	0.8545	0.824	0.7332	0.4771	-0.007	0.0431	0.1335	0.2281
Likes	0.7212	0.8318	0.4169	0.8545	1	0.8221	0.7442	0.4898	-0.0043	-0.0237	0.1149	0.1208
AddToCart	0.7092	0.8066	0.4161	0.824	0.8221	1	0.8925	0.5835	0.0025	-0.097	0.1525	0.1154
InCart	0.6308	0.7256	0.3705	0.7332	0.7442	0.8925	1	0.5579	0.0421	-0.0968	0.1361	0.0954
UnitsSold	0.4043	0.471	0.2336	0.4771	0.4898	0.5835	0.5579	1	0.2931	-0.0964	0.1259	0.0602
SalesIDR	0.0026	0.007	-0.003	-0.007	-0.0043	0.0025	0.0421	0.2931	1	0.0098	0.0173	-0.003
Price	0.0596	0.0414	0.0602	0.0431	-0.0237	-0.097	-0.0968	-0.0964	0.0098	1	-0.0159	0.114

Selected features (correlation >= 0.4 with UnitsSold):
['UnitsSold', 'AddToCart', 'InCart', 'Likes', 'SearchClicks', 'PageViews', 'Visits']

Figure 5. Pearson Correlation Matrix of Selected Predictors

The matrix indicates that the engagement measures are related not only to Units Sold but also to one another. Variables closer to purchase, particularly cart activity, have the strongest relationships with the target, while broader traffic measures such as Visits are weaker. These signals may help sellers recognize short-term demand, although their usefulness for inventory decisions still needs confirmation with other stores and under different promotional conditions.

3.3. Results of Prediction Models

Both models completed training under the shared configuration. Figure 6 provides one example of the BiLSTM training and validation losses for Product Code 10445853847. The two curves fall sharply at the beginning and remain relatively close afterward. Model comparison, however, is based on the held-out chronological validation results in Table 4 rather than on the appearance of a single loss curve.

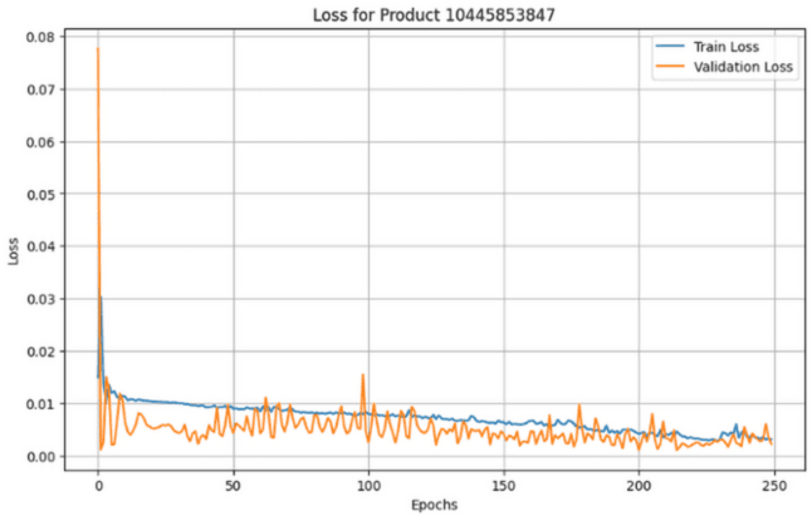


Figure 6. Loss Graph for Product 10445853847

Figure 7 compares actual and predicted weekly sales for Product Code 20843571485 during the hold-out period. The predictions reproduce several rises and fall in the series, but the largest gaps occur around sudden peaks. This pattern illustrates how difficult it is to forecast irregular demand when only a short weekly history is available.

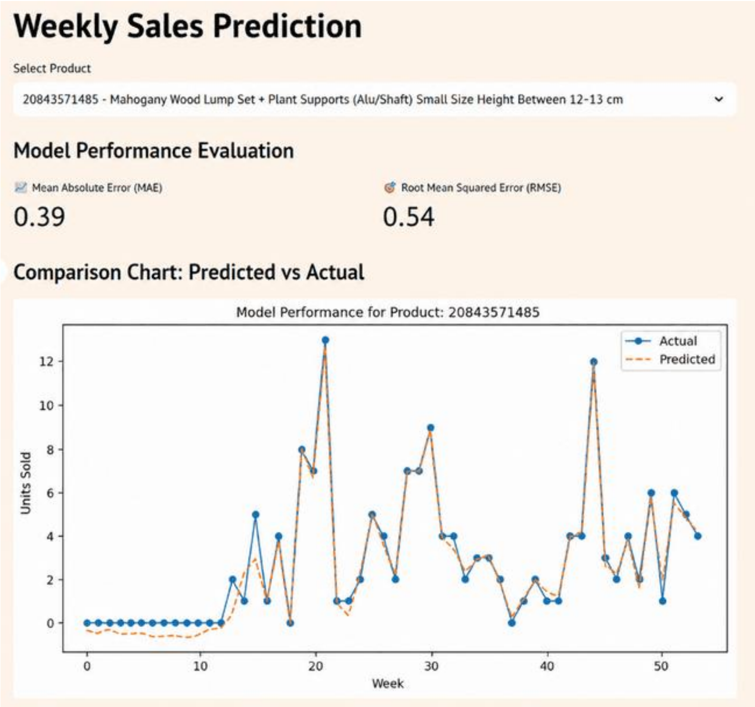


Figure 7. Results of Prediction Model for Product 20843571485

Table 4 presents the mean validation results for BiLSTM and LSTM across the ten products with the highest cumulative sales. MAE and RMSE are expressed as numbers of units sold. MAPE was calculated only for validation records whose actual sales were above zero.

Table 4. Average BiLSTM-LSTM Performance on the Chronological Validation Set

Model	Number of Product	MAE (Units)	RMSE (Units)	MAPE on Non-Zero Actuals (%)
BiLSTM	10	1.8505	2.4611	71.18
LSTM	10	1.7462	2.3977	69.57

On average, LSTM returned the smaller errors. Its MAE was 1.7462 units, compared with 1.8505 units for BiLSTM, and its RMSE was 2.3977 units, compared with 2.4611 units. The same pattern appeared in non-zero MAPE, which was 69.57% for LSTM and 71.18% for BiLSTM. Looking at individual products, LSTM recorded the lower MAE in eight of the ten cases. RMSE results were evenly split, with each architecture performing better for five products. BiLSTM was therefore competitive, but the experiment does not show a consistent advantage over the one-directional LSTM.

The difference between the two averages was modest, suggesting that the reverse pass did not add a dependable benefit for these product histories. Some series contain many weeks with little or no demand, abrupt peaks, and different numbers of validation samples. In such settings, a more complex architecture is not necessarily more accurate. Another thing to consider is that percentage errors also need to be contextualized. If the real sales are one or two units, then saying “we’re off by one unit” results in a very big percentage error. The MAE and the RMSE in units are therefore easier to interpret than the MAPE for stock planning.

3.4. Forecasting Results

Forecasts are given for horizons ranging from one to twelve weeks. For example, the following six weeks’ projections for the Mahogany Wooden Mortar are 6, 5, 5, 6, 9 and 4 units respectively (see Figure 8). It can help shape the conversation for purchases but the longer the horizon the more uncertain it gets. These are to be considered estimations, not guaranteed demand.

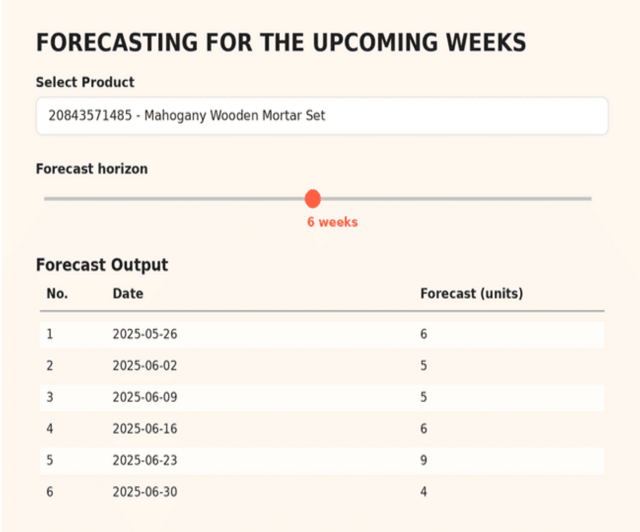


Figure 8. Six-Week Forecast Output

A seller can look at the forecast along with current stock and supplier lead time when deciding which products need attention. Higher forecasts can lead to an earlier evaluation of the reorder amount while a series of low forecasts might lead to conservative replenishment or to a consideration of promotional possibilities. The dashboard does not calculate safety stock or reorder points so the output is best used as a first pass tool rather than a full inventory policy.

3.5. Strategy for the Upcoming Week

The Strategy for the Upcoming Week page organizes products by their one week forward predictions. The result is a ranked list of outputs which allows the store owner to browse through a large number of product models without opening each result.

a. Top-Selling Products

The top-selling list ranks products that have the biggest estimates for next week. Figure 9: Predicted sales of Miniature Pinisi Ship and Mahogany Wooden Mortar are 10 and 6 units respectively. These items are highlighted for an early stock review, but the advice should still be validated against available inventory and supplier lead time before any purchase is made.

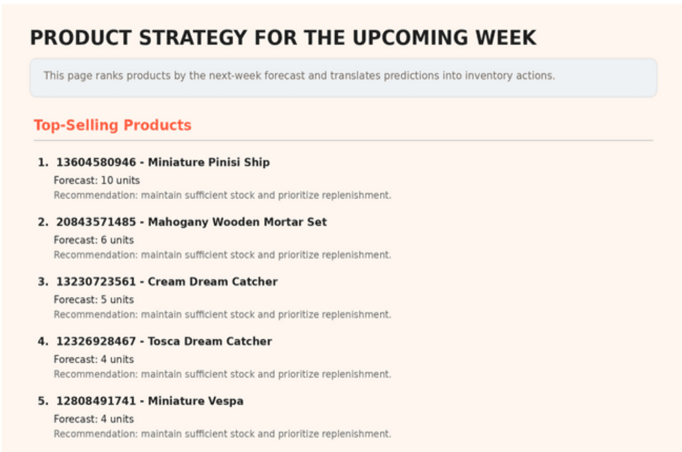


Figure 9. Top-Selling Product Strategy

b. Low-Selling Products

The low selling list is for products with zero or extremely small next-week estimates. For example, in Figure 10, the Wooden Owl Decoration is forecast to sell zero units. The dashboard suggests choosing a targeted campaign and avoiding needless replenishing. A zero-point forecast doesn't preclude no sales.

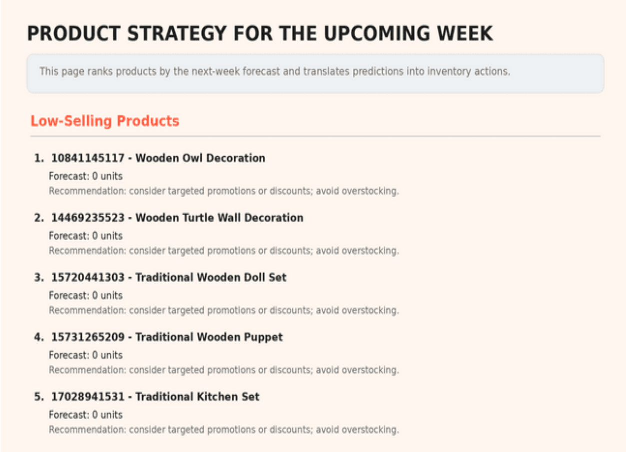


Figure 10. Low-Selling Product Strategy

Together the two lists link the model forecasts to a weekly routine stock assessment. The proposed actions are not hard and fast regulations and have to be understood in the context of the store's present business situation.

3.6. Study Limitations

The results should be interpreted with the limitations of the study in mind. All records are from one Shopee store and cover a two-year period only, so performance may be different for other sellers, product groups or marketplace platforms. The models also don't consider things such as promotions, holidays, competitive prices, supplier lead times or demand that may have disappeared when an item was out of stock. Since a distinct model was created for each product, many training sets were small and could have resulted in unstable estimations. The experimental benchmark was restricted to a unidirectional LSTM only and neither architecture was optimized via systematic hyper-parameter search. We did not evaluate ARIMA, naive forecasting, linear regression, gradient boosting and other alternative baselines. These restrictions are a limitation of the generalizability of the findings and on the robustness of any argument that one model is better than another.

4. Conclusion

This study combined correlation-based predictor selection, product-level sequence forecast, an LSTM benchmark, and a Streamlit dashboard for weekly inventory review. The strongest retained relationship of the cart activity was with Units Sold. The LSTM had a lower average MAE (1.7462 units) compared to the BiLSTM (1.8505 units) and a lower average RMSE (2.3977 against 2.4611 units) across the ten products utilized in the controlled comparison. BiLSTM performed similarly, but the bidirectional structure did not consistently improve the performance on this dataset. The main practical result is a dashboard that relates real marketplace behavior data to product projections and stock-prioritization cues. Further research should test new statistical and machine-learning baselines, rigorously modify hyperparameters, add promotion and calendar variables, and analyze the workflow on numerous retailers and marketplace platforms.

References

[1] A. Aziz, Mufarizzaturrizkiyah, and Leliya, Perilaku Gaya Hidup Konsumtif Mahasiswa Muslim E-Commerce: Survey pada Mahasiswa IAIN Syekh Nurjati Cirebon. Cirebon, Indonesia, 2020.

[2] T. Muhayat and S. Jayanta, "Prediksi harga smartphone menggunakan algoritma multiple linear regression," 2022.

[3] A. Gunawan, "Hubungan persediaan dengan penjualan pada laporan keuangan Koperasi Bukit Muria Jaya tahun 2021-2022," Jurnal Pengembangan Bisnis dan Manajemen, 2023.

[4] A. T. Nurani, A. Setiawan, and B. Susanto, "Perbandingan kinerja regresi decision tree dan regresi linear berganda untuk prediksi BMI pada dataset asthma," Jurnal Sains dan Edukasi Sains, vol. 6, no. 1, pp. 34-43, May 2023, doi: 10.24246/juses.v6i1p34-43.

[5] M. Sobron et al., Implementasi Artificial Intelligence pada Sistem Manufaktur Terpadu. 2021.

[6] M. Yang and J. Wang, "Adaptability of financial time series prediction based on BiLSTM," Procedia Computer Science, pp. 18-25, 2022, doi: 10.1016/j.procs.2022.01.003.

[7] Y. Syakir et al., "Analisis marketplace Shopee untuk memprediksi penjualan dengan algoritma regresi linier," 2022.

[8] N. Afrianto, D. H. Fudholi, and S. Rani, "Prediksi harga saham menggunakan BiLSTM dengan faktor sentimen publik," Jurnal RESTI, vol. 6, no. 1, pp. 41-46, Feb. 2022, doi: 10.29207/resti.v6i1.3676.

[9] D. I. Puteri, "Implementasi Long Short-Term Memory (LSTM) dan Bidirectional Long Short-Term Memory (BiLSTM) dalam prediksi harga saham syariah," Euler: Jurnal Ilmiah Matematika, Sains dan Teknologi, vol. 11, no. 1, pp. 35-43, May 2023, doi: 10.34312/euler.v11i1.19791.

[10] D. I. Puteri, G. Darmawan, and B. N. Ruchjana, "Prediksi harga saham syariah menggunakan Bidirectional Long Short-Term Memory (BiLSTM) dan algoritma grid search," Jambura Journal of Mathematics, vol. 6, no. 1, pp. 39-45, Feb. 2024, doi: 10.37905/jjom.v6i1.21199.

[11] R. A. Putra and E. Nurmawati, "Prediction-based stock portfolio optimization using Bidirectional Long Short-Term Memory (BiLSTM) and LSTM," Scientific Journal of Informatics, vol. 11, no. 3, pp. 609-620, Aug. 2024, doi: 10.15294/sji.v11i3.5941.

[12] N. Karin, G. Darmawan, and T. Hendrawati, "Enhancing PM2.5 prediction in Kemayoran District," BAREKENG: Journal of Mathematics and Its Applications, vol. 19, no. 1, pp. 185-198, 2025, doi: 10.30598/barekengvol19iss1pp0185-0198.

[13] K. Kwanda, D. E. Herwindiati, and M. D. Lauro, "Perbandingan LSTM dan Bidirectional LSTM pada sistem prediksi harga saham berbasis website," R2J, vol. 7, no. 1, 2024, doi: 10.38035/rrj.v7i1.

[14] I. Ubaedi and Y. M. Djaksana, "Optimasi algoritma C4.5 menggunakan metode forward selection dan stratified sampling untuk prediksi kelayakan kredit," Sistem Informasi, vol. 9, no. 1, 2022.

[15] Y. Karyadi and H. Santoso, "Prediksi kualitas udara dengan metode LSTM, Bidirectional LSTM, dan GRU," JATISI: Jurnal Teknik Informatika dan Sistem Informasi, 2022.

[16] V. Putri, "Normalisasi data menggunakan Min-Max untuk klasifikasi nasabah potensial pada pembelian properti dengan algoritma K-nearest neighbor," Informasi dan Teknologi Ilmiah, 2024.

[17] J. Irnawati and S. Samcay, "Pengaruh kompensasi terhadap kinerja karyawan pada PT Gratia Prima Indonesia," Jurnal Mandiri, vol. 4, no. 1, pp. 1-13, Jun. 2020, doi: 10.33753/mandiri.v4i1.101.

[18] F. M. Sari, R. N. Hadiati, and W. P. Sihotang, "Pearson correlation analysis of total population and number of motorized vehicles in Jambi Province," Multiproximity, vol. 2, no. 1, 2023, doi: 10.22437/multiproximity.v2i1.25568.

[19] D. J. Andriansyach, S. Sarwido, and H. Mulyo, "Short-term cryptocurrency price prediction using Bi-LSTM method with interactive web," Jurnal Teknologi dan Sistem Informasi Bisnis, vol. 6, no. 4, pp. 845-850, Nov. 2024, doi: 10.47233/jteksis.v6i4.1645.

[20] N. M. I. Jauhari, R. Wulanningrum, and A. B. Setiawan, "Sistem deteksi kendaraan menggunakan Streamlit dan metode YOLO," 2024.

[21] Y. Ahmadov and P. Helo, "Deep learning-based approach for forecasting intermittent online sales," Discover Artificial Intelligence, vol. 3, Art. no. 45, 2023, doi: 10.1007/s44163-023-00085-1.

[22] S. Mejía and J. Aguilar, "A demand forecasting system of product categories defined by their time series using a hybrid approach of ensemble learning with feature engineering," Computing, vol. 106, pp. 3945-3965, 2024, doi: 10.1007/s00607-024-01320-y.

[23] A. Mahmoud and A. Mohammed, "Leveraging hybrid deep learning models for enhanced multivariate time series forecasting," Neural Processing Letters, vol. 56, Art. no. 223, 2024, doi: 10.1007/s11063-024-11656-3.

[24] M. A. Jahin, A. Shahriar, and M. A. Amin, "MCDFN: Supply chain demand forecasting via an explainable multi-channel data fusion network model," Evolutionary Intelligence, vol. 18, Art. no. 66, 2025, doi: 10.1007/s12065-025-01053-7.

[25] Z. Zeng, W. Yang, M. Wang, M. Zhu, et al., "Chain drugstore sales prediction method based on the fusion of self-attention mechanism and LightGBM," Journal of Systems Science and Systems Engineering, 2025, doi: 10.1007/s11518-025-5652-1.