A Hybrid LSTM–MLP Model for Time-Series Sales Prediction

Onesinus Saut Parulian[1,*], Hilman Ferdinandus Pardede[1,2]

[1] *Department of Information Technology,* Universitas Nusa Mandiri*, Jakarta, Indonesia*

[2] *Research Center for Artificial Intelligence and Cybersecurity, National Research and Innovation Agency (BRIN), Jakarta, Indonesia*

[*] *Corresponding author: 14230003@nusamandiri.ac.id*

Abstract

Sales forecasting is an important part of an organization. This research builds a hybrid model combining Long Short-Term Memory (LSTM) with Multi-Layer Perceptron (MLP) to predict weekly retail sales. LSTMs are used to capture time-dependent patterns. However, LSTMs have limitations in capturing patterns or events that are not related to the time sequence (time-independent/non-temporal data). Therefore, this research combines MLP with LSTM to complement the capture of events or patterns that are not well captured by LSTM. Walmart Recruiting Store Sales Forecasting data is used with feature engineering and preprocessing to prepare for model training and evaluation. To measure the performance of the hybrid model, five metrics were used: Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), Mean Squared Error (MSE), Mean Absolute Percentage Error (MAPE), and the coefficient of determination (R^2). This LSTM-MLP hybrid model showed better performance with MAE of 2.39, RMSE of 3.39, MSE of 14.07, MAPE of 12.39, and R^2 approaching 1. This model outperformed other architectures such as the standalone MLP, GRU, and transformer models in terms of predicting weekly retail sales on the dataset tested in this study. This performance indicates that the hybrid LSTM-MLP method is effective in capturing sales patterns. This provides the potential for application in other fields or with other data.

Keywords: Sales prediction, LSTM, MLP, Hybrid model, Time series forecasting

1. Introduction

In supporting strategic decisions, artificial intelligence is becoming increasingly important, particularly in predicting sales (Venkataramanan et al. 2024). One of the traditional statistical models that has been used for a long time is ARIMA. Although it is less effective when dealing with non-linear and complex patterns (Wang 2024). Another alternative is offered by the deep learning LSTM model, which is more robust in capturing temporal dependencies (Liu 2024; Sonata and Heryadi 2024). Transformer-based architecture has shown its effectiveness in improving sales forecasting accuracy (Sun and Li 2024). However, the computational burden becomes more significant and training and testing times are longer. LSTMs and their variations have been successfully applied even with limited historical data in sectors such as agriculture and finance (Gyamerah and Korda 2021; Li and Wei 2023; Murugesan, Mishra, and Krishnan 2022). LSTM's ability to capture long-term temporal patterns outperforms traditional approaches like ARIMA in various case studies, including Gross Domestic Product (GDP) and government spending forecasting (Hamiane et al. 2024; Sautomo and Pardede 2021). Standalone LSTM faces challenges with time-independent patterns or events. On the other hand, MLP offers an approach to each data point independently (Weytjens, Lohmann, and Kleinsteuber 2021; Zhu et al. 2021). Improving model performance in sales prediction using a hybrid model compared to a standalone model can be achieved using hybrid methods such as LSTM-MLP (Ali et al. 2023; Raizada and Saini 2021; Shohan, Faruque, and Foo 2022; Tasdelen and Sen 2021; Yao 2023).

Therefore, this research proposes a hybrid LSTM-MLP model applied to Walmart Store Sales Forecasting data. Previous research has been conducted on the same data using a transformer-based model and has achieved excellent results (MSE 25.76 ± 2.47, RMSE 5.08 ± 1.23) (Sun and Li 2024). Despite the drawbacks, which include a more complex process, longer time required, and higher computational load. This hybrid approach is intended to be an alternative to the previously used methodological approaches. The evaluation will be conducted using the MAE, MSE, RMSE, and R^2 metrics. The results on the dataset show better performance (MSE 14.07, RMSE 3.39, MAE 2.39, MAPE 12.39), indicating the potential of this hybrid architecture for sales prediction needs.

2. Dataset

Walmart Recruiting: Store Sales Forecasting was used in this study. The data is accessible on Kaggle. Weekly data from February 2010 to October 2012, totaling 420,212 weekly data points, was tested. This weekly sales data is categorized by store and department. There are other columns such as temperature, fuel prices, promotions, consumer price index, unemployment rates, and holiday indicators. Explanation of each feature are in Tables 1, Table 2 and table 3.

Table 1 Data description of **features.csv**

No	Feature	Description
1	Store	Store ID
2	Date	Weekly Date
3	Temperature	Temperature
4	Fuel price	Fuel price
5	MarkDown1	Ongoing price reduction 1
6	MarkDown2	Ongoing price reduction 2
7	MarkDown3	Ongoing price reduction 3
8	MarkDown4	Ongoing price reduction 4
9	MarkDown5	Ongoing price reduction 5
10	CPI	Consumer Price Index
11	Unemployment	Unemployment Number
12	isHoliday	Is this week a holiday/special week?

Table 2 Data description of **stores.csv**

No	Feature	Description
1	*Store*	Store ID
2	*Type*	Store Type (A, B, C) divided by size of each store
3	*Size*	Store Size

Table 3 Data description of **train.csv**

No	Fitur / Attribute	Description
1	*Store*	Store ID
2	*Dept*	Department ID
3	*Date*	Weekly Date
4	*Weekly Sales*	Total Weekly Sales
5	*isHoliday*	Is this week a holiday/special week?

The data was divided into three parts in this study for the training, validation, and testing processes. Each for 113 weeks (331,742 samples), 15 weeks (44,192 samples), and 15 weeks (44,278 samples), respectively. The separation is done chronologically or by date to ensure that the model is learned and evaluated in a time series sequence. Table 4 provides a summary of the data distribution.

Table 4 Training, Validation and Testing Data Split

Data	Samples	Time Period	Weeks
Training	331,742	2010-02-05 s/d 2012-11-01	113

Validation	44,192	2012-04-06 s/d 2012-07-13	15
Test	44,278	2012-07-20 s/d 2012-10-26	15
Total	420,212	2010-02-05 s/d 2012-10-26	143

3. Research Methodology

We conducted an experimental method in this study after designing the LSTM-MLP hybrid model. An evaluation process was also carried out to improve the model's performance in predicting weekly retail sales. The research process consists of data collection, data processing, feature engineering, data splitting for training purposes, evaluation and testing, as well as model building and performance evaluation. Fig 1 visualizes the research stages.

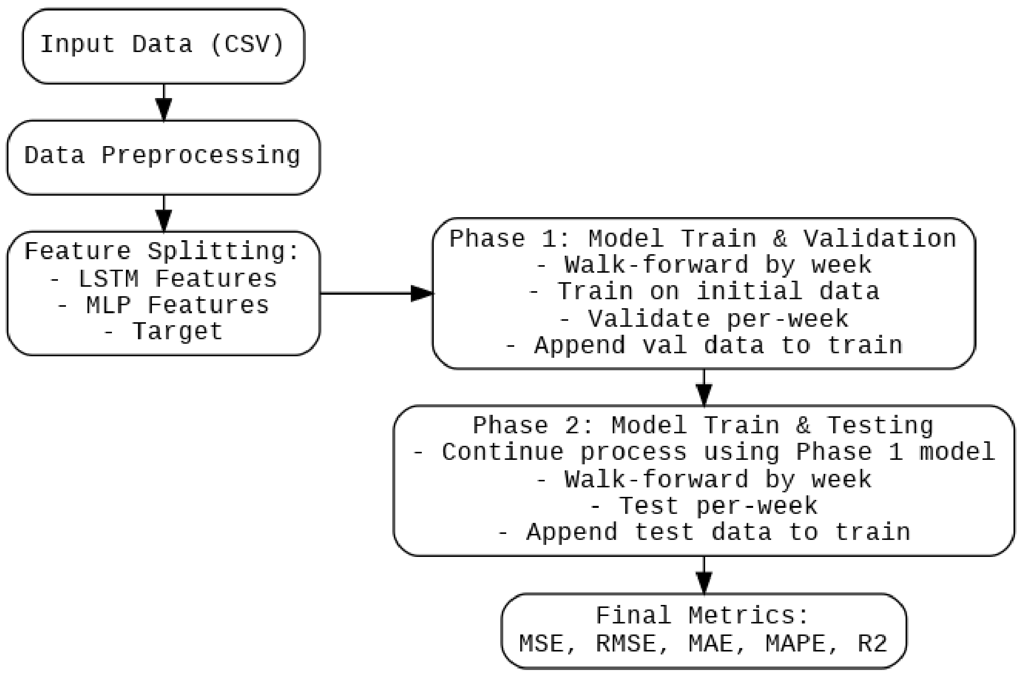

Fig 1. Research stages

3.1 Data Preprocessing

There are three different CSV files that need to be merged first to have complete data. In this process, we performed preprocessing by merging the three files based on the store and date columns. For missing values, we performed data imputation processes similar to those used for the CPI and unemployment features, using median values. Negative values, such as those in the markdown feature, were replaced with the number 0. The date feature was converted from a string data type to a date data type and format. All data was sorted chronologically by date. For outliers, data exclusion was performed by setting a z-score threshold of 2.5. All negative sales values were excluded from subsequent processes and steps.

3.2 Feature Engineering

We performed feature engineering so that the model could better capture relevant patterns in the data. Several new features were created, such as Year, Month, Week, and DayOfWeek, which were extracted from the Date feature. This aims to enable the model to understand annual context, monthly seasonal patterns, weekly analysis, and even differentiate between days and weeks. Another new feature is Lag_1_Week_Sales and Lag_2_Week_Sales, allowing the model to learn from sales 1 and 2 weeks prior. This enables the model to understand the influence of sales in each week on sales in the previous week. Other patterns or trends the model can learn from include the average sales over the past few weeks. Therefore, the features Rolling_Mean_2_Weeks and Rolling_Mean_4_Weeks were created, allowing the model to learn from the average of the last 2 and 4 weeks for each weekly sales data being analyzed and predicted. Both long-term and short-term trends can be studied with these new features. The combination of two features, such as store size and weekly sales, was created to see the influence between one feature and another. Therefore, we created a new feature called Store_Size_Interaction, which combines Size with Weekly Sales. The complete list of features and their distribution across each model can be found in Table 5. Time-related feature groups will be processed by LSTM, while independent, categorical, and numerical features will be processed by MLP.

Table 5 Feature splitting to be processed of each model

No	Features	LSTM	MLP
1	*Store*	X	✓
2	*Date*	X	X
3	*Dept*	X	✓
4	*Weekly Sales*	Target	
5	*Type*	X	✓
6	*Size*	X	✓
7	*Temperature*	X	✓
8	*Fuel price*	X	✓
9	*CPI*	X	✓
10	*Unemployment*	X	✓
11	*isHoliday*	✓	X
12	*Year*	✓	X

13	*Month*	✓	X
14	*Week*	✓	X
15	*DayOfWeek*	✓	X
16	*Max*	✓	X
17	*Min*	✓	X
18	*Mean*	✓	X
19	*Median*	✓	X
20	*Std*	✓	X
21	*Total_MarkDown*	✓	✓
22	*Lag_1_Week_Sales*	✓	X
23	*Lag_2_Week_Sales*	✓	X
24	*Rolling_Mean_2_Weeks*	✓	X
25	*Rolling_Mean_4_Weeks*	✓	X
26	*Store_Size_Interaction*	X	✓
27	*Size_x_Fuel*	X	✓
28	*CPI_x_Unemp*	X	✓
29	*Fuel_Price_sq*	X	✓

3.3 Hybrid Model

PyTorch is used for experiments and implementing the hybrid method. The model is built with two separate inputs, each processed by an LSTM and an MLP. Fifteen time-related features are processed by the LSTM, and thirteen independent or time-unbound features are processed by the MLP. The LSTM block is designed with a hidden layer size of 64. The MLP block is designed with two fully connected layers, the first with a size of 32 to map the input to 32 features, using the ReLU activation function, and the second fully connected layer with a size of 16 to reduce the features from 32 to 16. With this hybrid model design, LSTM is intended to capture sequential relationships in time series data, while MLP is intended to capture patterns or events that occur without time constraints (non-sequential data). The outputs of both processes are combined and passed to a fully connected layer to generate a regression prediction output. This final fully connected layer consists of 64 from the LSTM plus 16 from the MLP, which are then concatenated and mapped to 32 features. The final output of the 32 features is combined into a single output result representing the weekly sales regression prediction. Fig 2 illustrates the hybrid modeling architecture.

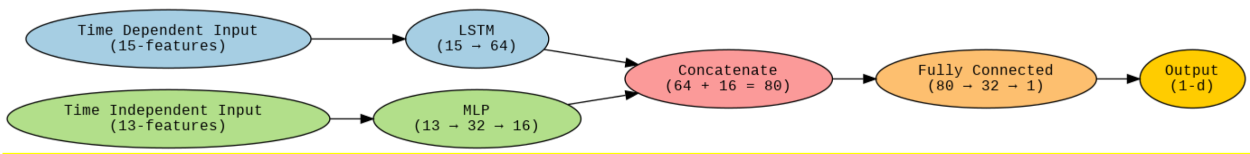

Fig 2. Hybrid Model Architecture proposed

The overall process is illustrated in Fig 3. This block diagram depicts the process from data merging to data processing, feature extraction, modeling, and finally, prediction and evaluation.

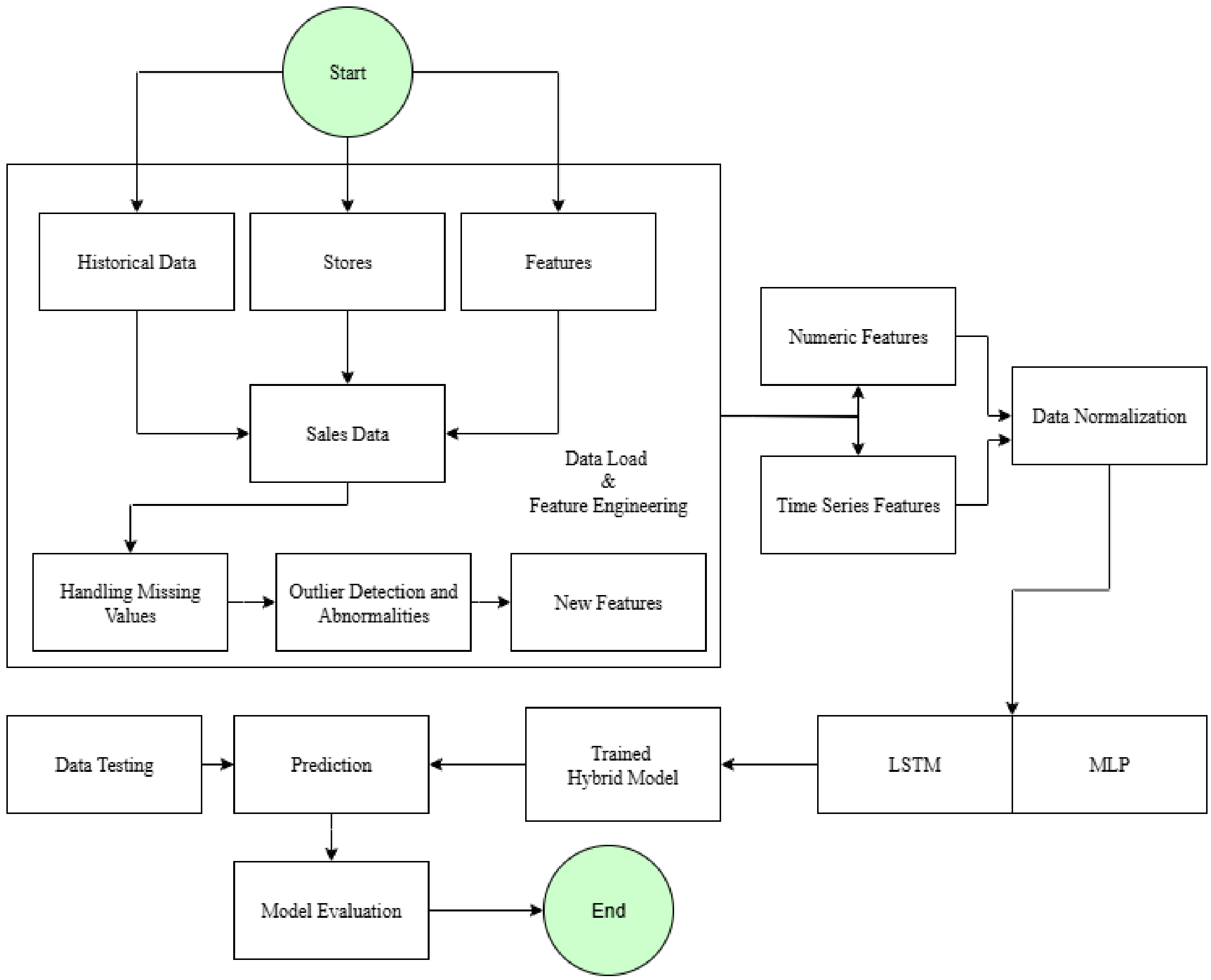

Fig 3. Block diagram of hybrid LSTM and MLP

We divided the training phase into two phases. Each phase uses a walk-forward method where the model is trained iteratively on continuously expanding training data and validated and tested each week. The process continues by shifting one week from the validation and testing set to training. The first phase of training and validation was conducted using 113 weeks of training data, which was continuously added to with validation data from week 1 to week 15. Then, in the second phase, training and evaluation were carried out, with the model and training data continuing from the previous phase, resulting in a total of 128 weeks of data being trained, with an addition of 1 week from the evaluation data from week 1 to week 15. The modeling strategy is illustrated in Fig. 4.

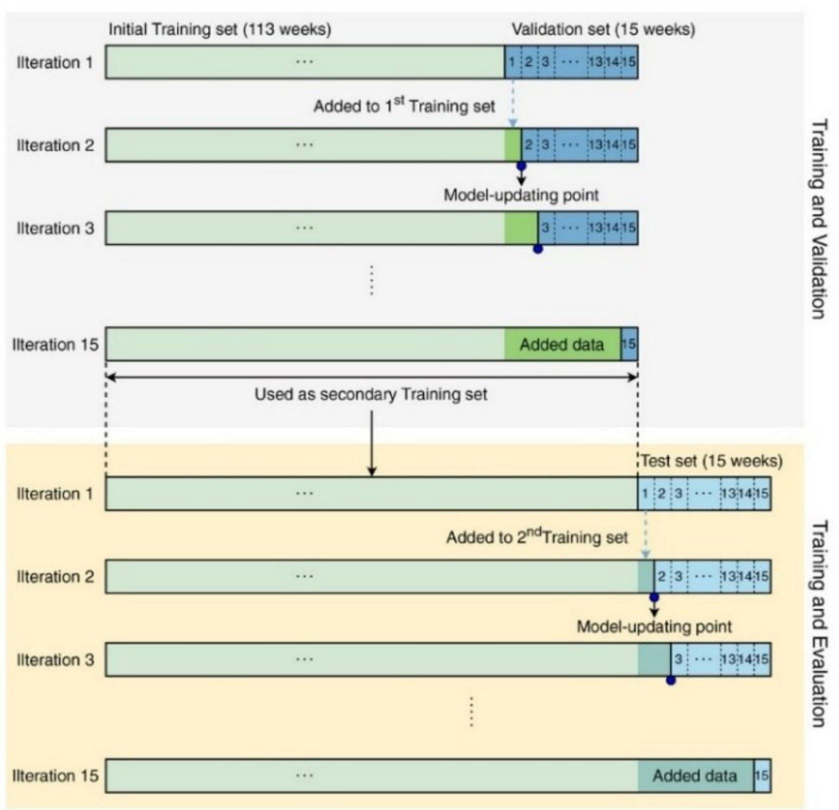

Fig 4. Modeling strategy (Sun and Li 2024)

3.4 Model Evaluation

This research applies the Adam optimizer, with a batch size set to 32. The model's performance is evaluated using five metrics: Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), and the coefficient of determination (R^2). In the process of evaluating these metrics, they were calculated by averaging 15 weeks of iterations, where iteration 1 had 15 weeks of sales test data, down to iteration 15 with only 1 week of sales test data. With evaluation periods ranging from just 1 week to 15 weeks, the model's performance can be assessed in predicting both short-term and long-term time series data.

4. Results and Discussion

4.1 Device Information

The specifications of the devices we used in this study are in Table 6.

Table 6 Device specification during research

Device	Google Compute Engine backend (GPU)
Runtime Type	Python3
Hardware Accelerator	T4 GPU
System RAM	12.7 GB
GPU RAM	15.0 GB
Disk	112.6 GB

4.2 Modeling

The combination of LSTM and MLP models was designed as separate layers before being combined and passed to the fully connected layer and output layer. Table 7 and Fig 5 provide detailed information regarding the implementation of the hybrid model.

Table 7 LSTM and MLP Hybrid modeling

Output	1 prediction value (regression)
Optimizer	Adam
Learning Rate	0.001
Loss Function	Mean Squared Error (MSE)
Batch Size	32
Epoch	1 (per week iteration)
Validation Scheme	Walk-forward based on weekly data

```
=================================================================
Layer (type:depth-idx)                   Param #
=================================================================
LSTMMLP                                  --
├─LSTM: 1-1                              20,736
├─Sequential: 1-2                        --
│    └─Linear: 2-1                       448
│    └─ReLU: 2-2                         --
│    └─Dropout: 2-3                      --
│    └─Linear: 2-4                       528
│    └─ReLU: 2-5                         --
├─Sequential: 1-3                        --
│    └─Linear: 2-6                       2,592
│    └─ReLU: 2-7                         --
│    └─Linear: 2-8                       33
=================================================================
Total params: 24,337
Trainable params: 24,337
Non-trainable params: 0
=================================================================
```

Fig 5. Model architecture of hybrid LSTM and MLP

The explanation of the implementation of the hybrid LSTM and MLP model is detailed in Table 8.

Table 8 LSTM and MLP Hybrid modeling detail explanation

<table>
<tr><td>Model</td><td>- Model: LSTMMLP
- Main Component: LSTM, MLP, Fully Connected (FC)</td></tr>
<tr><td colspan="2">Initializing a hybrid model that combines LSTM for temporal sequences and MLP for non-sequential feature processing. The goal is to combine the strengths of both in a single model.</td></tr>
<tr><td>Layer LSTM</td><td>- Input Size: lstm_input_size
- Hidden Size: 64
- Batch First: True</td></tr>
<tr><td colspan="2">LSTM is used to process sequential data, with a hidden layer size of 64 to capture sequence information. The goal is to understand patterns in sequential data (such as time data).</td></tr>
<tr><td>Input to LSTM</td><td>- Input: lstm_input
- Dimension: (batch_size, seq_len, features)
- Unsqueeze is used to add dimension</td></tr>
<tr><td colspan="2">The sequential input is dimensioned to the appropriate format and then passed to the LSTM for processing. The goal is to ensure that the sequential data is processed correctly by the LSTM.</td></tr>
<tr><td>Output LSTM</td><td>- Output Dimension: (batch_size, seq_len, 64)
- Last Timestep: Taken for features (dimensions: [batch_size, 64])</td></tr>
<tr><td colspan="2">The LSTM output contains a sequence representation from each timestep, but only the output from the last timestep is selected for use as the primary feature. This aims to capture the most relevant information from the entire sequence.</td></tr>
<tr><td>Layer MLP</td><td>- Input Size: mlp_input_size
- Layer: 2 Linear Layers
- Activation: ReLU
- Dropout: 0.2</td></tr>
<tr><td colspan="2">MLP is used to process additional features unrelated to the time sequence, with two layers: Linear and Dropout to avoid overfitting. The goal of MLP is to extract additional features useful for prediction.</td></tr>
</table>

<table>
<tr><td>Input to MLP</td><td>- Input: mlp_input
- Input Dimension: (batch_size, mlp_input_size)</td></tr>
<tr><td colspan="2">The input to the MLP is processed through a multilayer perceptron network to produce features of size 16. The goal of the MLP is to extract and process additional information that is independent of the order of the data.</td></tr>
<tr><td>Features Combining</td><td>- LSTM Feature: [batch_size, 64]
- MLP Feature: [batch_size, 16]
- Combined Features: [batch_size, 80]</td></tr>
<tr><td colspan="2">Features extracted from the LSTM (64 dimensions) and MLP (16 dimensions) were combined into a single tensor of size 80 for further processing. The goal was to combine information from the two sources (LSTM and MLP) into a single, more comprehensive feature set.</td></tr>
<tr><td>Layer Fully Connected (FC)</td><td>- Input Size: 80 (LSTM + MLP Input)
- Layer: 2 Linear Layers
- Activation: ReLU
- Output: 1 (for regression)</td></tr>
<tr><td colspan="2">The fully connected layer is used to process the combined features into a single final value (output). The final output is a single number that can be used for regression or classification, depending on the task at hand.</td></tr>
<tr><td>Model Output</td><td>- Output: [batch_size, 1] (scalar output)</td></tr>
<tr><td colspan="2">The model output is one value per batch, which is used for prediction in regression tasks. The ultimate goal is to generate predictions based on the information processed by the LSTM and MLP.</td></tr>
</table>

We conducted experiments using a date-based approach in two main phases to simulate prediction scenarios in a real-world context. Phase 1 (validation walk) is an iterative training and validation process. The model is trained based on initial training data and then validated on data from each week based on validation data. In each iteration, the data from that week is added to the training set for the next iteration (forward validation). Phase 2 (Testing Walk) is the next step where the model results from Phase 1 are used to test the final performance on test data. Similar to the validation phase, testing is conducted in stages.

4.3 Experiment Result

We compared several standalone models with a hybrid model. Standalone LSTM and MLP were also tested and compared. The results showed that the hybrid LSTM and MLP method produced better results compared to other tested methods and previous approaches. The comparison results can be seen in Table 9.

Table 9 Experiment result

Model	**MSE**	**RMSE**	**MAE**	**MAPE**	**R2**
MLP-based model	24314.76 ± 1831.35	154.34 ± 5.73	101.34 ± 4.10	444.16 ± 41.14	0.9885 ± 0.000804
GRU-based model	721.9958 ± 121.98	25.04 ± 2.51	18.55 ± 2.12	13.95 ± 0.91	0.9997 ± 0.000057
LSTM-based model	78.76 ± 18.60	7.94 ± 1.01	6.90 ± 0.98	12.04 ± 1.10	1.0000 ± 0.000008
LSTM-GRU model	336.48 ± 60.09	17.02 ± 1.76	11.49 ± 1.29	18.91 ± 2.51	0.9998 ± 0.000029
GRU-LSTM model	234.38 ± 68.24	13.24 ± 1.98	8.18 ± 1.26	34.09 ± 4.51	0.99 ± 0.000031
GRU-MLP model	3711.27 ± 542.04	58.23 ± 4.61	41.75 ± 4.22	60.50 ± 10.62	0.99 ± 0.000262
Transformer Based (Sun and Li 2024)	25.76 ± 4.65	5.08 ± 1.23	3.12 ± 0.37	9.49 ± 1.11	0.95 ± 0.08
LSTM-MLP (Ours)	**14.07 ± 2.9**	**3.39 ± 0.41**	**2.39 ± 0.37**	**12.39 ± 3.04**	**1.00 ± 0.000001**

We present a comparison of four error metrics for the models tested in Fig 6.

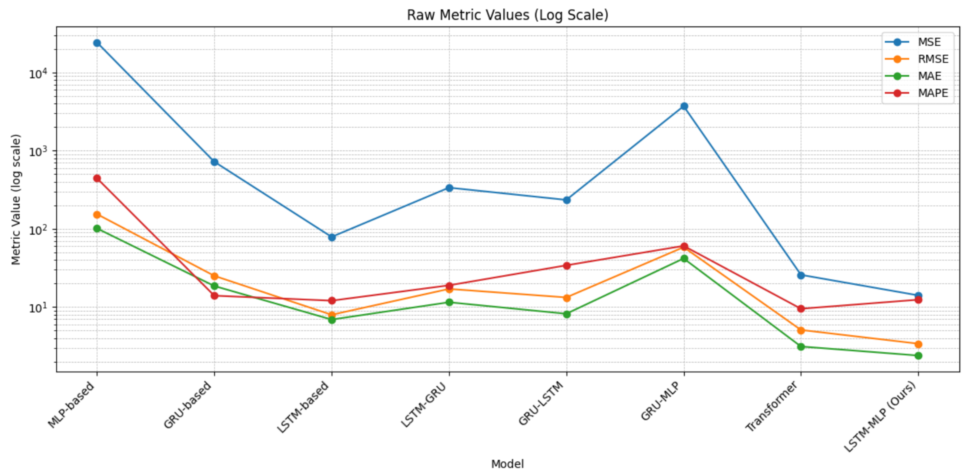

Fig 6. Error comparison among tested methods

Fig 7 visualizes weekly sales values against absolute error values.

Fig 7. Absolute Error based on actual sales value prediction

It can be seen that all the absolute error values tend to be low, namely lower than 20, although there are still a small number of results with errors in the range of 20-30, specifically in weekly sales data with a value of 6000 - 7000. This is because the data in that range is only 14 thousand data points compared to other data ranges that have more training data, such as the 1000-5000 range with a total of almost 100 thousand data points. However, the error value is still relatively small, making this very acceptable. A comparison of the mean absolute error results among the models can be seen in Fig 8.

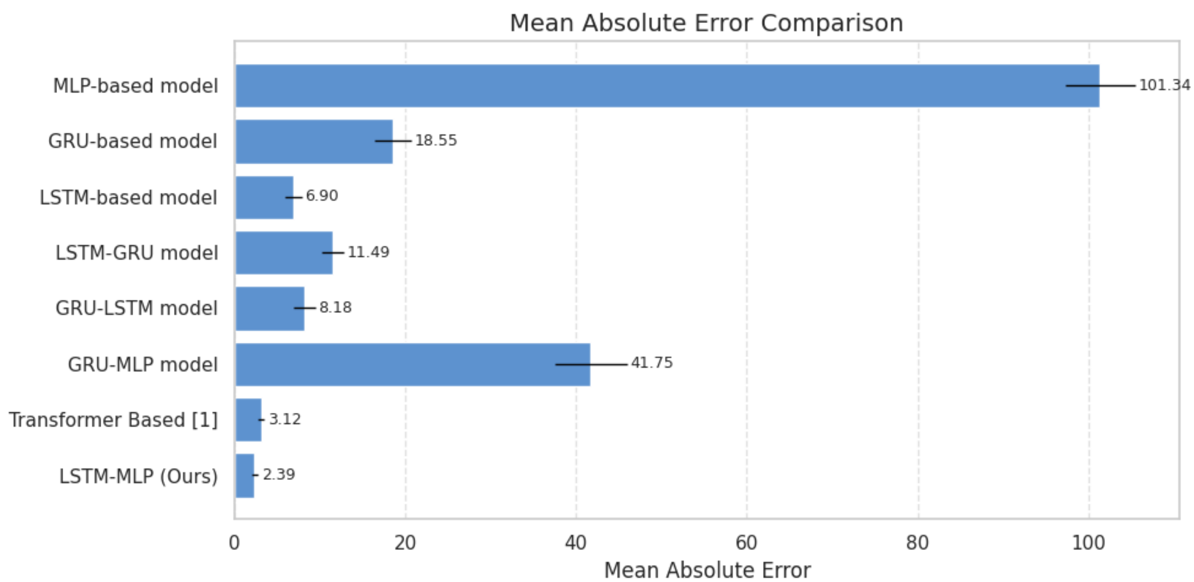

Fig 8. MAE comparison among tested methods

4.4 Discussion

An analysis was conducted for each store and department to assess the model's performance across different data categories. Fig 9 provides an overview of the location of the highest error and the stores and departments with the lowest error values.

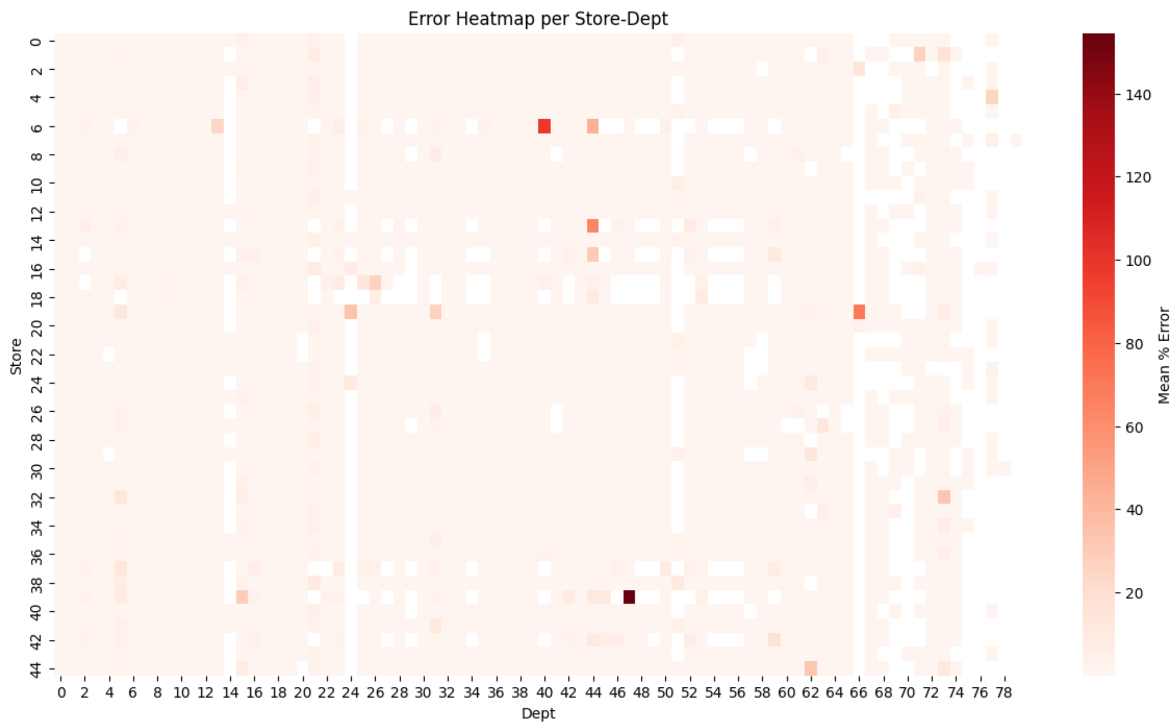

Fig 9. Heatmap error per department per store

It can be seen that the prediction error rate for store ID 39 and department ID 47 is very high. This is because only 10 data points are available in total, with very small, even negative, target values. This is very insufficient and good, causing the model to be unable to learn and predict the specifics of this store and department well compared to data in other departments where an average of more than 140 data points are available for each store. Table 10 provides data information on store 39 and department 47.

Table 10 Department in a store with the lowest error

#	Store	Dept	Date	Weekly_Sales
1	39	47	2/12/2010	40
2	39	47	8/27/2010	34
3	39	47	12/3/2010	25
4	39	47	12/31/2010	-79
5	39	47	3/11/2011	-15
6	39	47	5/6/2011	15
7	39	47	12/9/2011	-14.4
8	39	47	1/20/2012	-19
9	39	47	2/10/2012	19
10	39	47	6/22/2012	18

For more details on the size of the influence of each feature, see Fig. 10. The red color indicates that the feature has a high influence on the model, while blue indicates a low influence.

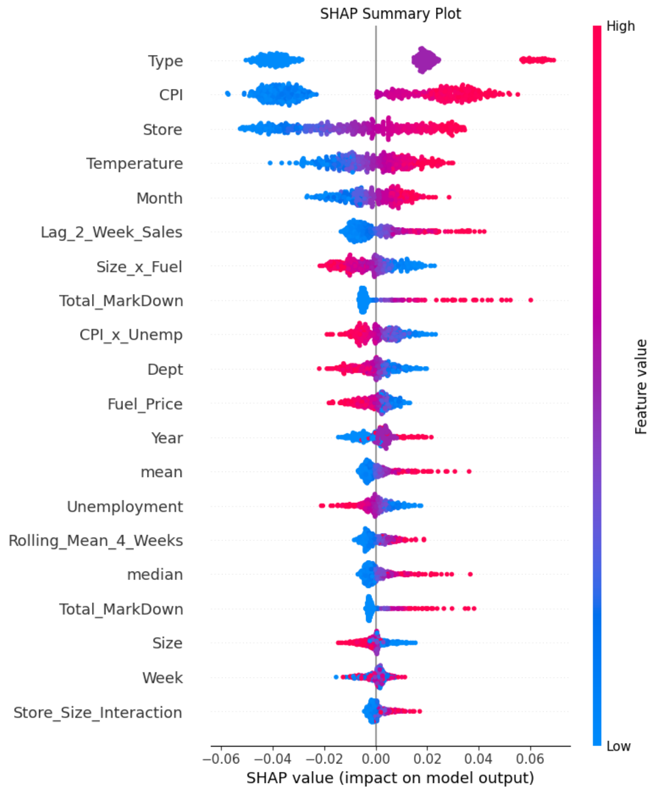

Fig 10. SHAP value of each feature and the impact to the model

5. Conclusions

The research we have conducted shows that a combination of deep learning models such as hybrid LSTM and MLP performs better in predicting weekly retail sales. Testing and comparison of individual deep learning models. This hybrid approach successfully produced low error values. These results indicate the superiority of LSTM in processing time series data and capturing sales transaction patterns, especially over the long term. MLP can complement LSTM in capturing patterns and events that influence the model but are not time-dependent. Including very rare events caused by independent factors. Therefore, this approach can be considered for further research and regression implementation in similar case studies. Further research can test this hybrid LSTM and MLP approach on other domains and datasets. The characteristics of other datasets may differ, so adjusting settings such as hyperparameters may be necessary.

References

Ali, Muhammad, Dost Muhammad Khan, Huda M. Alshanbari, and Abd Al Aziz Hosni El-Bagoury. 2023. "Prediction of Complex Stock Market Data Using an Improved Hybrid EMD-LSTM Model." *Applied Sciences (Switzerland)* 13(3). doi:10.3390/app13031429.

Gyamerah, Seth, and Dennis Redeemer Korda. 2021. "Prediction of Stock Market Returns Using LSTM Model and Traditional Statistical Model." *International Journal of Computer Applications* 183(37):57–61. doi:10.5120/ijca2021921773.

Hamiane, Sana, Youssef Ghanou, Hamid Khalifi, and Meryam Telmem. 2024. "Comparative Analysis of LSTM, ARIMA, and Hybrid Models for Forecasting Future GDP." *Ingénierie Des Systèmes d Information* 29(03):853–61. doi:10.18280/isi.290306.

Li, Huiling, and Daijun Wei. 2023. "Prediction Based on Traditional Network Prediction Model and LSTM Deep Neural Network Model." Pp. 726–30 in *Proceedings of the 2023 4th International Conference on Computing, Networks and Internet of Things*. New York, NY, USA: ACM.

Liu, Yi. 2024. "Sales Forecasting Based on Transformer-LSTM Model." *Highlights in Science, Engineering and Technology* 85:776–82. doi:10.54097/qdavzg31.

Murugesan, R., Eva Mishra, and Akash Hari Krishnan. 2022. "Forecasting Agricultural Commodities Prices Using Deep Learning-Based Models: Basic LSTM, Bi-LSTM, Stacked LSTM, CNN LSTM, and Convolutional LSTM." *International Journal of Sustainable Agricultural Management and Informatics* 8(3):242. doi:10.1504/IJSAMI.2022.125757.

Raizada, Stuti, and Jatinderkumar R. Saini. 2021. "Comparative Analysis of Supervised Machine Learning Techniques for Sales Forecasting." *International Journal of Advanced Computer Science and Applications* 12(11):2021. doi:10.14569/IJACSA.2021.0121112.

Sautomo, Sabar, and Hilman Ferdinandus Pardede. 2021. "Prediksi Belanja Pemerintah Indonesia Menggunakan Long Short-Term Memory (LSTM)." *Jurnal RESTI* 5(1):99–106. doi:10.29207/resti.v5i1.2815.

Shohan, Md Jamal Ahmed, Md Omar Faruque, and Simon Y. Foo. 2022. "Forecasting of Electric Load Using a Hybrid LSTM-Neural Prophet Model." *Energies* 15(6). doi:10.3390/en15062158.

Sonata, Ilvico, and Yaya Heryadi. 2024. "Comparison of LSTM and Transformer for Time Series Data Forecasting." Pp. 491–95 in *2024 7th International Conference on Informatics and Computational Sciences (ICICoS)*. IEEE.

Sun, Yupeng, and Tian Li. 2024. "A Transformer-Based Framework for Enterprise Sales Forecasting." *PeerJ Computer Science* 10:1–14. doi:10.7717/peerj-cs.2503.

Tasdelen, Abdulkadir, and Baha Sen. 2021. "A Hybrid CNN-LSTM Model for Pre-MiRNA Classification." *Scientific Reports* 11(1). doi:10.1038/s41598-021-93656-0.

Venkataramanan, Srinivasan, Leeladhar Gudala, Ashok Kumar Reddy Sadhu, Amith Kumar Reddy, and Senior Engineering Manager. 2024. *Leveraging Artificial Intelligence for Enhanced Sales Forecasting Accuracy: A Review of AI-Driven Techniques and Practical Applications in Customer Relationship Management Systems*. Vol. 4. https://www.researchgate.net/publication/389945780.

Wang, Chenyu. 2024. "Stock Price Forecasting by ARIMA, Linear Regression, LSTM and Decomposition Linear Models." *Applied and Computational Engineering* 55(1):61–68. doi:10.54254/2755-2721/55/20241572.

Weytjens, Hans, Enrico Lohmann, and Martin Kleinsteuber. 2021. "Cash Flow Prediction: MLP and LSTM Compared to ARIMA and Prophet." *Electronic Commerce Research* 21(2):371–91. doi:10.1007/s10660-019-09362-7.

Yao, Bo. 2023. *Walmart Sales Prediction Based on Decision Tree, Random Forest, and K Neighbors Regressor*. Vol. 2022.

Zhu, Weiwei, Jinglin Wu, Ting Fu, Junhua Wang, Jie Zhang, and Qiangqiang Shangguan. 2021. "Dynamic Prediction of Traffic Incident Duration on Urban Expressways: A Deep Learning Approach Based on LSTM and MLP." *Journal of Intelligent and Connected Vehicles* 4(2):80–91. doi:10.1108/JICV-03-2021-0004.