Asia Pac. J. Math. 2024 **11**:65

OPTIMIZATION OF PERTAMAX PRICE PREDICTION ANALYSIS IN PERTASHOP WEST NUSA TENGGARA REGION USING DEEP LEARNING MODEL

SURYA IBRANI, ARIO YUDO HUSODO*, I GEDE PASEK SUTA WIJAYA

Master of Information Technology, University of Mataram, Mataram 83125, Indonesia

*Corresponding author: ario@unram.ac.id

Received May 26, 2024

Abstract. This research examines the implementation of the Pertamina Shop (Pertashop) program in West Nusa Tenggara, Indonesia, with a focus on analyzing changes in Pertamax fuel prices. This program aims to increase energy access, meet fuel needs, empower the local economy, and encourage regional economic development. Pertamax price fluctuations are a challenge for Pertashop, which has an impact on financial stability and customer satisfaction. This research proposes a solution using accurate price predictions with deep learning models, namely Recurrent Neural Network (RNN), Long Short-Term Memory (LSTM), and Gated Recurrent Unit (GRU). The research results show that the LSTM and GRU models with optimal hyperparameters provide more accurate Pertamax price predictions than RNN. The analysis uses Pertamina data, crude oil prices, and the USD-IDR exchange rate from January 2016 to October 2023. The LSTM model shows the best performance with training RMSE 0.0083 and testing RMSE 0.0084, and R^2 of 0.9989. This research provides valuable insight into the economic impact and sustainability of the Pertashop program at the local level, particularly in West Nusa Tenggara. It is hoped that these findings will be useful for policy makers, businesses and the public to better understand the impact of the program and potentially influence similar initiatives at the national level.

2020 Mathematics Subject Classification. 68T07; 68T09.

Key words and phrases. Pertamax price; Pertashop; deep learning model; West Nusa Tenggara Province.

1. Introduction

Villages, as independent entities, play an important role in the development and management of their communities. The success of a village in achieving self-sufficiency is reflected in the authority and financial support outlined in the Village Revenue and Expenditure Budget (Village Revenue and Expenditure Budget). Village involvement in implementing development programs or collaborating with other villages creates opportunities to improve community welfare. Recognizing the important

DOI: 10.28924/APJM/11-65

role of villages, the Indonesian government is encouraging collaboration between the Ministry of Home Affairs and PT Pertamina (Persero) through the Pertamina Shop (Pertashop) program to expand energy access, meet fuel needs, empower the community's economy, and encourage local economic development.

This research examines the implementation of the Pertashop Program in West Nusa Tenggara Province (NTB), a province that has a significant number of Pertashop outlets. In this context, relevant literature includes an analysis of changes in Pertamax prices as a key variable that influences people's purchasing power and regional economic growth [13]. Literature regarding Pertamax price fluctuations provides insight into its potential impact, especially in the context of changes in world crude oil prices. Although several studies have identified these factors, information regarding their specific impacts at the village level, particularly in NTB is still lacking. Therefore, the main aim of this research is to determine the implementation of the Pertashop Program in NTB, with a focus on analyzing changes in Pertamax prices. In addition to that, indeed this research target fills this knowledge gap and provides a better understanding of the economic impact and sustainability of the Pertashop Program at the local level. This offers a positive contribution to village development, especially in NTB, and provides valuable insights into similar programs on a national scale.

On the other hand, Pertamax price fluctuations imposed by the government provide a challenge for Pertashop. These unpredictable prices hinder Pertashop's business operations which implies negative impacts such as financial losses and decreased customer satisfaction. Consequently, research is necessary to overcome these challenges. One of the proposed solutions is an accurate analysis of Pertamax price predictions to help Pertashop anticipate unexpected price changes and make more appropriate decisions in managing its business. The beneficiaries of this research will generally be Pertashop, to remain competitive in the market and provide the best service to its customers. Therefore, it is important to analyze the price prediction for Pertamax at Pertashop outlets in NTB using a deep learning model.

The prediction of changes in world oil prices can be determined using machine learning [2], viewing price fluctuations as measurable variables. Several research has demonstrated the predictability of global crude oil prices through various methods, such as artificial neural networks [17], deep learning [5], and Long Short-Term Memory (LSTM) deep learning methods based on the transfer of previous knowledge [3] Similarly, AdaBoost-LSTM, AdaBoost-Gated Recurrent Unit (GRU) [2], and LSTM-based deep learning models have been applied to forecast world crude oil prices over specific periods. Furthermore, the combination of LSTM and Grey Forecasting (GM) models has shown the capability to predict Brent crude oil prices with higher accuracy compared to using LSTM or GM alone, achieving an accuracy of 92% [16].

Recently, research studies employing LSTM models have been developed in various global case studies, such as the use of LSTM in predicting groundwater levels. In this study, a high accuracy level was achieved, with the LSTM model accurately predicting groundwater levels for the next 10 years with a 95% accuracy rate [12]. Similar research has also been conducted on oil production from 16 oil wells in China over 5 years, from 2014 to 2018. The LSTM model developed in this study demonstrated high accuracy in predicting oil well performance for the upcoming period, with a reported accuracy of 96% [11]. In line with research utilizing LSTM to forecast oil production from two oil wells in Saudi Arabia, the LSTM model achieved a prediction accuracy of 95% for the first well and 92% for the second well for the subsequent period [12]. The data used in this study spans from 1998 to 2018, covering 20 years.

Based on the aforementioned studies, it is evident that deep learning methods can effectively predict outcomes. The upcoming research will focus on predicting Pertamax prices in Pertashop outlets in NTB, representing a specific and relatively unexplored research subject. This study can be positioned as an extension of prior research on crude oil price prediction using deep learning. Previous studies have generally concentrated on predicting crude oil prices in a general sense, often using only oil data as input variables. In contrast, this research specifically targets the prediction of Pertamax prices in Pertashop outlets in NTB by incorporating additional data such as crude oil prices and the exchange rate between the dollar and the rupiah as input variables for obtaining Pertamax price predictions. Furthermore, this research utilizes three types of deep learning models—RNN, LSTM, and GRU—which have not been extensively explored in previous studies. The analysis in this study includes hyperparameter variables such as epoch, learning rate, and batch size. By examining these variables, the aim is to provide insights into the hyperparameter values that yield the most optimal prediction results. The expected outcomes of this research are to offer insights into optimizing Pertamax price predictions in Pertashop outlets in NTB using deep learning, along with identifying the hyperparameter values (including epoch, learning rate, and batch size) that produce the most optimal results .

2. Materials and Methods

The data utilized in this research comprises Pertamax prices from January 2016 to October 2023 sourced from Pertamina's data, which in this research took data on Pertamax prices in NTB. Several variables are included in the dataset including crude oil prices, the dollar-to-rupiah exchange rate, and Pertamax prices. The obtained data is then presented in .csv format to facilitate the data processing process. The acquired dataset consists of four columns and 1981 rows. The price data can be seen in Table 2.1.

TABLE 2.1. **Price Data**

No.	Date	Price Oil	Price IDR	Pertamax Price
1	1/4/2016	36.76	13915.0	9000
2	1/5/2016	35.97	13862.5	9000
3	...	...	...	...
4	10/30/2023	82.31	15885.0	14000
5	10/31/2023	81.02	15880.0	14000

2.1. **Data processing.** The first step of processing consists of normalizing the data using a min-max scaler. The data will be normalized using Min-Max Scaler to ensure that the data falls within the range of 0 and 1 [6] with the following formula:

$$x_{scaled} = \frac{(x - x_{min})}{x_{max} - x_{min}}.$$

Variable xscaled is the result of data normalization, x is the original value of the variable that we want to scale, xmin represents the smallest value of the variable, and xmax represents the largest value of the variable. The results of data normalization are depicted in Figure 1.

FIGURE 1. Data Normalization

2.2. **Methods.** Recurrent Neural Network (RNN) is a form of artificial neural network highly suitable for identifying patterns in sequential data, such as text, video, speech, language, genomes, and time series data. RNN is a powerful algorithm capable of performing classification, clustering, and prediction on data, especially in the context of time series and text. RNN can be seen as a Multi-Layer Perceptron (MLP) with the addition of loops in its architecture. In RNN, when the sequence is long enough, the gradient (crucial for adjusting weights and biases) is computed during the training process (backpropagation). This gradient can experience vanishing (multiplying many small values below 1) or exploding (multiplying many large values above 1), which can lead to very slow model training [1].

2.1 Recurrent Neural Network (RNN)

RNN is a type of artificial neural network structure where the process is iteratively repeated to handle input, typically sequential data [8]. RNN falls into the category of deep learning as it involves processing data through multiple layers. The characteristics of sequential data involve processing samples sequentially [14], such as in the time dimension, where each sample in the sequence has a close relationship with the preceding one [1]. An example of RNN architecture can be seen in Figure 2.

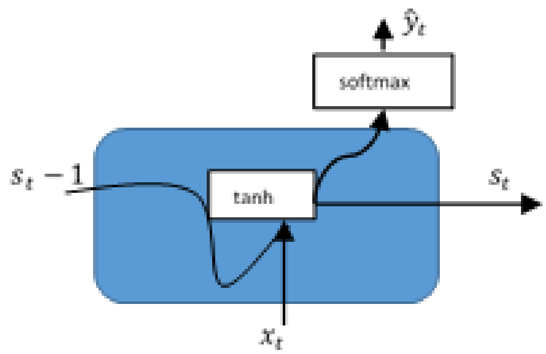

FIGURE 2. Recurrent Neural Network (RNN) Architecture

Note that x_t is input Processing by RNN and y_t produces an output st in each processing step. Meanwhile, RNN stores an internal state, which is passed from one-time step to the next st-1 is the previous state [19]

2.2 Long Short-Term Memory Algorithm (LSTM)

Long Short-Term Memory (LSTM) is one type of RNN capable of processing long and complex sequential data [1]. RNNs typically use backpropagation models for learning, but long chains in the RNN topology can lead to vanishing gradients [4]. LSTM is a type of RNN developed to address the issue of vanishing gradients [15]. It is modified by incorporating a memory cell, allowing it to store information over a longer period [19]. LSTM has the advantage of a memory block that can determine the values to be considered as relevant output for a given input. The following image shows an example of an LSTM architecture.LSTM architecture can be seen in Figure 3.

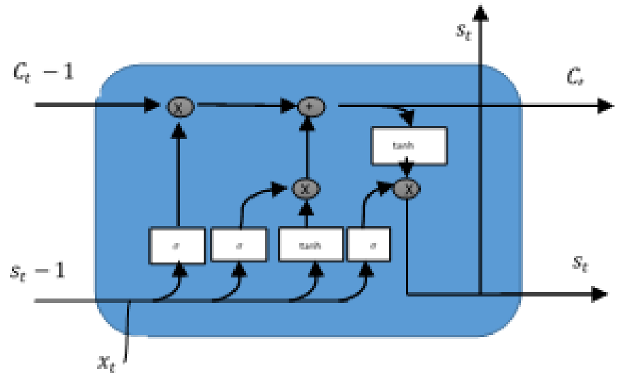

Figure 3. Long Short-Term Memory (LSTM) Architecture

In this case, C_t is the cell state or memory cell, σ is a sigmoid gate, and tan_h is an activation function [9]. The LSTM structure consists of three main parts: the input layer, the output layer, and the hidden layer [12]. The hidden layer itself is composed of memory cell units, each of which has three gates: the input gate, the forget gate, and the output gate [11].

2.3 Gated Recurrent Unit (GRU)

The advantage of GRU lies in its simpler computational process compared to LSTM [18], despite being capable of achieving comparable accuracy and remaining effective in addressing the issue of vanishing gradients [17]. Suppose that C_t is cell state or memory cell, σ is a sigmoid gate, and tan_h is an activation function [10]. Then, GRU architecture is presented below Figure 4.

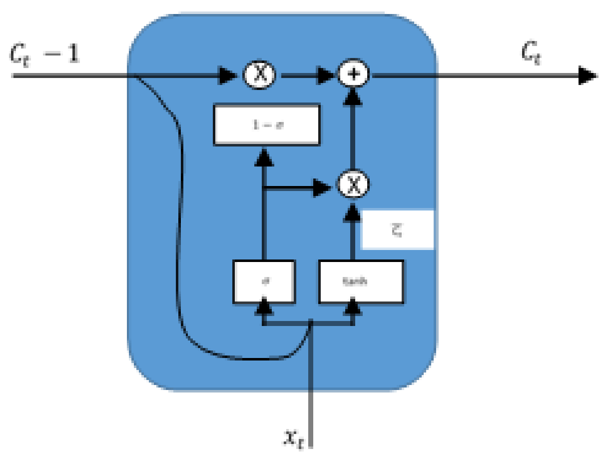

Figure 4. Gated Recurrent Unit (GRU) Architecture

2.3. **Model Evaluation.** Model evaluation involves the use of several indices, with Root Mean Square Error (RMSE) and R^2 being employed to assess model performance [7]. RMSE represents the standard deviation of the model's prediction results. It is a measure of how accurately a model predicts values. RMSE is calculated by taking the square root of the Mean Square Error (MSE). The RMSE value can range from 0 to ∞. The accuracy of error estimation methods can be measured by the RMSE value. A small RMSE indicates that the model can produce values close to the observed ones. Therefore, a model with a smaller RMSE is considered more accurate than one with a larger RMSE [1].

R^2 is a coefficient of determination ranging between 0 and 1. A value approaching 1 indicates that the independent variable can explain the dependent variable very well. Conversely, a value close to 0 suggests that the independent variable cannot explain the dependent variable effectively. In this study, prediction results will be evaluated using RMSE and R^2-values [1] with the calculation formula is as follows

$$RMSE = \sqrt{\frac{1}{n}\sum_{t=1}^{n}(A_t - F_t)^2}$$

where A_t is the true value, F_t is an approximate value, and n is the number of data points. Let $\bar{Y}$ be the mean value of the dependent variable, then we have

$$R^2 = 1 - \frac{\sum_{t=1}^{n}(A_t - F_t)^2}{\sum_{t=1}^{n}(A_t - \bar{Y})^2}$$

3. Results and Discussion

The deep learning models, RNN, LSTM, and GRU, were successfully trained using historical data on Pertamax prices, crude oil prices, and USD to IDR exchange rates to generate predictions for Pertamax prices. Their performance was evaluated using metrics such as Root Mean Squared Error (RMSE) and R^2. The results of this performance evaluation serve as the primary indicators of the model's performance in predicting prices. Each parameter will be tested, and the testing results of parameters that yield a sufficiently good loss will be used for the next round of testing to generate the best model.

Figure 5 illustrates the structural design of the method layer, representing the foundation for training the model detailed in Figure 6 within this study. Following the training process, the model's performance will be reflected through the depiction of loss and validation loss, showcased in Figure 7 via a graph illustrating training and validation loss.

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 lstm (LSTM)                 (None, 1, 64)             17408

 dropout (Dropout)           (None, 1, 64)             0

 lstm_1 (LSTM)               (None, 1, 64)             33024

 dropout_1 (Dropout)         (None, 1, 64)             0

 lstm_2 (LSTM)               (None, 1, 64)             33024

 lstm_3 (LSTM)               (None, 1, 64)             33024

 lstm_4 (LSTM)               (None, 64)                33024

 dense (Dense)               (None, 1)                 65

=================================================================
Total params: 149569 (584.25 KB)
Trainable params: 149569 (584.25 KB)
```

FIGURE 5. The architecture of the Method's Layer

```
# Train the model
history = model.fit(X_train, y_train, epochs=200, batch_size=32, validation_data=(X_test, y_test), verbose=2)

Epoch 1/200
50/50 - 11s - loss: 0.1557 - val_loss: 0.1459 - 11s/epoch - 219ms/step
Epoch 2/200
50/50 - 1s - loss: 0.1380 - val_loss: 0.1253 - 809ms/epoch - 16ms/step
Epoch 3/200
50/50 - 0s - loss: 0.1106 - val_loss: 0.0884 - 445ms/epoch - 9ms/step
Epoch 4/200
50/50 - 0s - loss: 0.0629 - val_loss: 0.0457 - 470ms/epoch - 9ms/step
Epoch 5/200
50/50 - 0s - loss: 0.0384 - val_loss: 0.0294 - 461ms/epoch - 9ms/step
```

FIGURE 6. Train the model

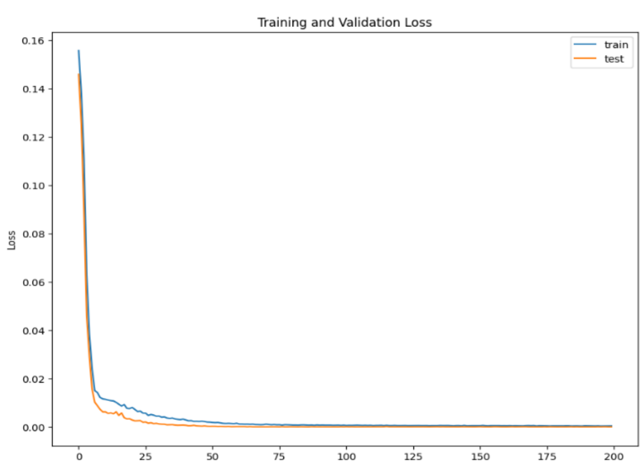

FIGURE 7. Training and validation loss

From the provided configuration of the model, training will be conducted with the given hyperparameters. Subsequently, we can examine the values of both training and validation loss from the trained model. The RNN, LSTM, and GRU methods, utilizing the hyperparameters from the best-performing models obtained during testing, will yield results upon their application. These results encompass a comparison between actual values and predicted values for both training and testing data.

Parameter Testing Stages, stage 1 data composition, stage 2 hidden layer, stage 3 dropout, stage 4 batch size, stage 5 Epoch, stage 6 Learning rate. As can be seen in Table 3.1, the results of the hyperparameter testing stage of partition data using the RNN model. In Table 3.2 and Table 3.3 you can see the results of changing the hyperparameters of the RNN method using layer and drop out parameters. The hyperparameter value that provides the best results will be selected for use in the next step, seen in Table 3.4 using the batch size parameter. For Table 3.5 and Table 3.6 using the epoch and learning rate parameters. So that the RNN model being tested will use the optimal value of each given hyperparameter.

Table 3.1. **RNN model data partition parameter testing stage**

No.	Data Partition	Loss	Val_Loss	RMSE (training)	RMSE (testing)	R^2
1	50 : 50	$3.05e-04$	$2.72e-04$	0.0169	0.0165	0.9961
2	65 : 35	$4.19e-04$	$3.67e-04$	0.0200	0.0191	0.9947
3	70 : 30	$3.48e-04$	$2.90e-04$	0.0181	0.0170	0.9957
4	80 : 20	$3.41e-05$	$3.32e-05$	0.0057	0.0057	0.9995
5	90 : 10	$1.79e-04$	$1.66e-04$	0.0129	0.0129	0.9976

Table 3.2. **RNN model layer parameter testing stage**

No.	Layer	Loss	Val_Loss	RMSE (training)	RMSE (testing)	R^2
1	3 layer	$4.89e-06$	$4.10e-06$	0.0021	0.0020	0.9999
2	4 layer	$3.16e-06$	$2.88e-06$	0.0016	0.0016	0.9999
3	5 layer	$5.98e-06$	$4.74e-06$	0.0023	0.0021	0.9999
4	6 layer	$2.97e-06$	$2.87e-06$	0.0016	0.0016	0.9999
5	7 layer	$3.39e-06$	$3.29e-06$	0.0017	0.0018	0.9999

Table 3.3. **RNN model drop out parameter testing stage**

No.	Drop Out	Loss	Val_Loss	RMSE (training)	RMSE (testing)	R^2
1	1	0.0016	0.0039	0.0635	0.0624	0.9437
2	2	0.0030	0.0062	0.0796	0.0785	0.9109
3	3	0.0069	0.0024	0.0491	0.0486	0.9658

Table 3.4. **RNN model batch size parameter testing stage**

No.	Batch Size	Loss	Val_Loss	RMSE (training)	RMSE (testing)	R^2
1	16	0.0053	0.0028	0.0543	0.0531	0.9593
2	24	0.0060	0.0012	0.0356	0.0346	0.9826
3	32	0.0068	0.0012	0.0348	0.0345	0.9828
4	40	0.0063	0.0027	0.0529	0.0519	0.9610
5	48	0.0057	0.0029	0.0544	0.0535	0.9586
6	56	0.0081	0.0051	0.0710	0.0701	0.9289
7	64	0.0089	0.0057	0.0762	0.0754	0.9178

Table 3.5. **RNN model epoch parameter testing stage**

No.	Epoch	Loss	Val_Loss	RMSE (training)	RMSE (testing)	R^2
1	50	0.0068	0.0022	0.0480	0.0471	0.9679
2	80	0.0045	0.0029	0.0557	0.0542	0.9576
3	100	0.0045	0.0024	0.0498	0.0489	0.9654
4	120	0.0033	0.0043	0.0668	0.0652	0.9385
5	150	0.0035	0.0043	0.0666	0.0652	0.9385
6	180	0.0036	0.0028	0.0544	0.0526	0.9600
7	200	0.0027	0.0031	0.0566	0.0555	0.9555

Table 3.6. **RNN model learning rate parameter testing stage**

No.	Learning rate	Loss	Val_Loss	RMSE (training)	RMSE (testing)	R^2
1	0.0001	0.0033	0.0049	0.0714	0.0699	0.9294
2	0.0002	0.0032	0.0029	0.0549	0.0540	0.9578
3	0.0003	0.0025	0.0049	0.0718	0.0702	0.9288
4	0.0004	0.0021	0.0015	0.0394	0.0385	0.9786
5	0.0005	0.0021	0.0047	0.0700	0.0688	0.9317

Based on the results of the experiment for data partitioning in the RNN, the obtained loss on testing is considered underfitting because the training loss is larger than the validation loss. To obtain the best model from the first experiment, the one with the smallest difference between training RMSE and testing RMSE is selected. Based on the initial test, the optimal combination of data is found in the second experiment, with a training RMSE of 0.0200 and a testing RMSE of 0.0191. The R^2-value obtained is 0.9947. Subsequently, further experiments were conducted to optimize hyperparameters such as the number of layers, dropout, batch size, epochs, and learning rate.

From the hyperparameter experimentation on the learning rate in RNN, for the second and fourth experiments, the obtained loss on testing is classified as underfitting because the training loss is greater than the validation loss. Meanwhile, for the first, third, and fifth experiments, testing is considered overfitting as the training loss is smaller than the validation loss. The best model from the fourth experiment is selected based on the smallest difference between training RMSE and testing RMSE.

In the fifth experiment, focusing on the learning rate hyperparameter, the optimal combination of data from the trained dataset is found in the fourth experiment, with a learning rate of 0.0004. The obtained values are 0.03948 for training RMSE and 0.03948 for testing RMSE. The R^2-value obtained is 0.9786.

By using the RNN method, the following results are obtained, data partitioning was obtained by dividing 65% of the data for training and 35% for testing. The hidden layer was obtained using 3 hidden layers with 50 neurons each. Dropout was obtained using 3 dropouts with a value of 0.2. The batch size was obtained using a batch size of 32. The epoch was obtained using 200 epochs. The learning rate was obtained using 0.0004.

With the hyperparameters of the best model obtained during testing using the RNN method, the total RMSE (training) is 0.0508 and the total RMSE (testing) is 0.0497. The R^2 value obtained is 0.9643.

By carrying out the same method as hyperparameter testing in the RNN method, we use it with the LSTM and GRU methods, we will get the most optimal results for each stage used. In this study, the LSTM method utilizes the best hyperparameters. The data partitioning results in an 80-20 split, with 80% for training data and 20% for testing data. The hidden layers consist of 5 layers with 64 neurons each. Dropout is implemented with 2 dropout layers, each having a dropout rate of 0.2. The batch size is set to 32, and the training process involves 200 epochs. The learning rate is configured to be 0.0002.

With the hyperparameters of the best model obtained during testing using the LSTM method, the total RMSE (training) is 0.0083 and the total RMSE (testing) is 0.0084. The R^2 value obtained is 0.9989, indicating a very good quality.

The utilization of the GRU method in this study involves the application of the best hyperparameters. The data is partitioned with a split of 65 data points for training and 35 for testing. The hidden layers consist of 5 layers, each containing 64 neurons. Dropout is implemented with 1 dropout layer and a dropout rate of 0.2. The batch size is set to 16, and the training process involves 180 epochs. The learning rate is configured to be 0.0003.

With the hyperparameters of the best model obtained during testing using the GRU method, the total RMSE (training) is 0.0076 and the total RMSE (testing) is 0.0080. The R^2-value obtained is 0.9990, indicating a very good quality. The results for the metrics can be observed in Table 3.7.

Table 3.7. **Comparison Table of Metrics for Different Model Approaches**

No.	Method	RMSE (training)	RMSE (testing)	R^2
1	RNN	0.0508	0.0497	0.9643
2	LSTM	0.0083	0.0084	0.9989
3	GRU	0.0076	0.0080	0.9990

Based on the table above, the LSTM and GRU models for making predictions provide much better performance than the RNN model, this is indicated by significantly lower RMSE values and much higher R^2 for both models. If we look at the LSTM and GRU models the performance of these two models is almost the same, but GRU has a slight advantage in terms of RMSE on test data. Although the difference is small, it shows that the GRU model is slightly better at generalizing to previously unseen data.

The R^2 value is close to 1 for the LSTM and GRU models indicating that the two models are able to explain very large variations in the data very well. In this research, the use of the LSTM or GRU model is more recommended than the RNN model in making predictions because of its better performance. If you want to prioritize performance, then the LSTM model can be chosen because it is slightly superior to the difference between RMSE training and RMSE testing with an R^2 value close to 1.

After optimizing the hyperparameters in the LSTM model for Pertamax price predictions, Pertashop partners can integrate the optimized LSTM model into their platform to provide more accurate Pertamax price estimates to customers. Pertashop partners can adjust their pricing strategies dynamically, thus this LSTM model not only provides better price estimates, but also increases operational efficiency and profits for Pertashop.

4. Conclusions

This research demonstrated Pertamax price prediction based on the analysis of deep learning methods, including RNN, LSTM, and GRU. The study covers Pertamax prices from January 2016 to October 2023, utilizing data from Pertamina and incorporating other variables such as crude oil prices and USD to IDR exchange rate. Moreover, we presented the identification of optimal hyperparameter values, including epoch, learning rate, and batch size to ensure the best performance of each model. The RNN model achieved optimal performance with a training RMSE of 0.0508 and testing RMSE of 0.0497, yielding an R^2 value of 0.9643. The LSTM model, with hyperparameters such as 5 hidden layers, 64 neurons, and a learning rate of 0.0002, exhibited exceptional results with a training RMSE of 0.0083, testing RMSE of 0.0084, and an impressive R^2 value of 0.9989. Similarly, the GRU model, configured with 5 hidden layers, 64 neurons, and a learning rate of 0.0003, demonstrated excellent performance with a training RMSE of 0.0076, testing RMSE of 0.0080, and an outstanding R^2 value of 0.9990. The results indicate that the application of the identified hyperparameters, particularly in LSTM and GRU models, produces the optimal price predictions compared to the RNN method. Optimization of epoch, learning rate, and batch size significantly improves prediction accuracy. The GRU model with optimized hyperparameters can be applied and used by Pertashop partners to provide more accurate Pertamax price estimates in the NTB region.

Authors' Contributions

All authors have read and approved the final version of the manuscript. The authors contributed equally to this work.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

References

[1] S. Bouktif, A. Fiaz, A. Ouni, M. Serhani, Optimal deep learning LSTM model for electric load forecasting using feature selection and genetic algorithm: Comparison with machine learning approaches, Energies 11 (2018), 1636. https://doi.org/10.3390/en11071636.

[2] G.A. Busari, D.H. Lim, Crude oil price prediction: A comparison between AdaBoost-LSTM and AdaBoost-GRU for improving forecasting performance, Comp. Chem. Eng. 155 (2021), 107513. https://doi.org/10.1016/j.compchemeng.2021.107513.

[3] Z. Cen, J. Wang, Crude oil price prediction model with long short term memory deep learning based on prior knowledge data transfer, Energy 169 (2019), 160–171. https://doi.org/10.1016/j.energy.2018.12.016.

[4] S. Gao, Y. Huang, S. Zhang, J. Han, G. Wang, M. Zhang, Q. Lin, Short-term runoff prediction with GRU and LSTM networks without requiring time step optimization during sample generation, J. Hydrol. 589 (2020), 125188. https://doi.org/10.1016/j.jhydrol.2020.125188.

[5] M. Hussein, Y. Azhar, Prediksi harga minyak dunia dengan metode deep learning, Fount. Inform. J. 6 (2021), 2548–5113.

[6] C. Jörges, C. Berkenbrink, B. Stumpe, Prediction and reconstruction of ocean wave heights based on bathymetric data using LSTM neural networks, Ocean Eng. 232 (2021), 109046. https://doi.org/10.1016/j.oceaneng.2021.109046.

[7] A. Khumaidi, I.A. Nirmala, Algoritma long short term memory dengan hyperparameter tuning: Prediksi penjualan produk, Deepublish, Yogyakarta, 2022.

[8] Y. Li, H. Cao, Prediction for tourism flow based on LSTM neural network, Procedia Comp. Sci. 129 (2018), 277–283. https://doi.org/10.1016/j.procs.2018.03.076.

[9] X. Song, Y. Liu, L. Xue, J. Wang, J. Zhang, J. Wang, L. Jiang, Z. Cheng, Time-series well performance prediction based on Long Short-Term Memory (LSTM) neural network model, J. Pet. Sci. Eng. 186 (2020), 106682. https://doi.org/10.1016/j.petrol.2019.106682.

[10] M. Pan, H. Zhou, J. Cao, Y. Liu, J. Hao, S. Li, C.H. Chen, Water level prediction model based on GRU and CNN, IEEE Access 8 (2020), 60090–60100. https://doi.org/10.1109/access.2020.2982433.

[11] J. Qiu, B. Wang, C. Zhou, Forecasting stock prices with long-short term memory neural network based on attention mechanism, PLoS ONE 15 (2020), e0227222. https://doi.org/10.1371/journal.pone.0227222.

[12] A. Sagheer, M. Kotb, Time series forecasting of petroleum production using deep LSTM recurrent networks, Neurocomputing 323 (2019), 203–213. https://doi.org/10.1016/j.neucom.2018.09.082.

[13] M. Sari, G.J. Yanris, M.N.S. Hasibuan, Analysis of the neural network method to determine interest in buying pertamax fuel, SinkrOn 8 (2023), 1031–1039. https://doi.org/10.33395/sinkron.v8i2.12292.

[14] L. Sehovac, K. Grolinger, Deep learning for load forecasting: Sequence to sequence recurrent neural networks with attention, IEEE Access 8 (2020), 36411–36426. https://doi.org/10.1109/access.2020.2975738.

[15] F. Shahid, A. Zameer, M. Muneeb, Predictions for COVID-19 with deep learning models of LSTM, GRU and Bi-LSTM, Chaos Solitons Fractals 140 (2020), 110212. https://doi.org/10.1016/j.chaos.2020.110212.

[16] J. Veri, S. Surmayanti, G. Guslendra, Prediksi harga minyak mentah menggunakan jaringan syaraf tiruan, Matrik: J. Manaj. Tek. Inform. Rekayasa Komput. 21 (2022), 503–512. https://doi.org/10.30812/matrik.v21i3.1382.

[17] T. Yao, Z. Wang, Crude oil price prediction based on LSTM network and GM (1,1) model, Grey Syst. Theory Appl. 11 (2020), 80–94. https://doi.org/10.1108/gs-03-2020-0031.

[18] D. Zhang, M.R. Kabuka, Combining weather condition data to predict traffic flow: a GRU-based deep learning approach, IET Intell. Transp. Syst. 12 (2018), 578–585. https://doi.org/10.1049/iet-its.2017.0313.

[19] J. Zhang, Y. Zhu, X. Zhang, M. Ye, J. Yang, Developing a long short-term memory (LSTM) based model for predicting water table depth in agricultural areas, J. Hydrol. 561 (2018), 918–929. https://doi.org/10.1016/j.jhydrol.2018.04.065.