REVISTA DE NEGÓCIOS
Inovação e Estratégia em Organizações

Petrobras PETR3 stock price forecast: an application of CNN-LSTM neural networks

José Airton Azevedo dos Santos
Doutor em Engenharia Elétrica
Universidade Tecnológica Federal do Paraná (UTFPR)
airton@utfpr.edu.br

Leandro de Oliveira
Mestrando em Tecnologias Computacionais para o Agronegócio
Universidade Tecnológica Federal do Paraná (UTFPR)
leandrooliveira@alunos.utfpr.edu.br

Received 17.05.2024 - Reviewed 24.09.2024 - Accepted 23.03.2026
ISSN 1980 -4431 Double blind review

Petrobras PETR3 stock price forecast: an application of CNN-LSTM neural networks

Abstract

This work proposes using the CNN-LSTM network to predict the opening price of Petrobras PETR3 stock. The database offers a daily series of PETR3 stock prices between January/2012 and December/2022, totaling 2713 observations. Multivariate prediction models based on the CNN-LSTM (Convolutional Neural Network - Long Short Term Memory), LSTM (Long Short Term Memory), GRU (Gated Recurrent Unit), and CNN (Convolutional Neural Network) networks were implemented in the Python language. Results obtained from the four models were compared using the metrics RSME (Root Mean Squared Error), MAPE (Mean Absolute Percent Error), and MAE (Mean Absolute Error). It was verified for a 30-day horizon that the CNN-LSTM model presented a better performance.

Keywords: Deep learning. Keras. Python.

1 Introduction

Predicting stock market price movements is a very complex challenge, but it plays an essential role in the appropriate timing of buying or selling stocks (Khaidem et al., 2016). According to Lu et al. (2020), accurately predicting stock price change can reduce stock investment risk and improve investment return. In this sense, Artificial Neural Networks (ANNs) models can be used as support tools in decision-making.

Neural networks, computational techniques inspired by the neural structure of intelligent organisms, are tools of great importance in a planning system. They are considered a processing scheme capable of storing knowledge and making it available for a given application (Sebastian, 2016; Pinheiro et al., 2020; Bastiani et al., 2018; Haykin, 2001; Santos & Chaucoski, 2020; Yan et al., 2020). Deep-learning-based models have also been widely used in financial areas, such as forecasting stock price, portfolio optimization, risk management, financial information processing, and trade execution strategies (Yan et al., 2020).

Several works used models based on Deep-learning to predict stock prices. Chen et al. (2015) used the LSTM model to predict the price of shares on China's Shanghai and Shenzhen stock exchanges. Samarawickrama and Fernando (2017) used LSTM and GRU networks to predict the share price of three companies on the Colombo stock exchange in Sri Lanka. Althelaya et al. (2018) used Stacked LSTM and GRU models to predict stock prices. The LSTM and GRU architectures were compared using historical data from the S&P 500 index. Lu et al. (2020) proposed using the CNN-LSTM-AM (Attention) network to predict the closing stock price for 1000 trading days of the Shanghai Composite Index. It is observed that previous studies, in the analysis of financial time series, indicate that the use of deep learning models is successful.

In this context, this work proposes using the CNN-LSTM neural network to forecast the opening price of Petrobras PETR3 stock. PETR3 is the ticker for Petrobras common shares. A ticker is a code used on the stock exchange to identify and trade a specific asset. It is observed that this work uses a multivariate analysis. Unlike most time series forecasting techniques, which generally use a univariate approach. According to Widipruta et al. (2021), forecasting time series values using a univariate model, whether an econometric or a machine learning model, has some disadvantages. These disadvantages are related to the high noise and volatility of financial time series and the relationship between dependent and independent variables, which are subject to unpredictable changes over time.

The article is organized as follows. Section 2 presents the methodology used to predict the open price of PETR3 stocks. In Section 3, the results obtained from the application of this methodology are presented. The final comments and conclusions in Section 4 conclude the article.

2 Theoretical Framework

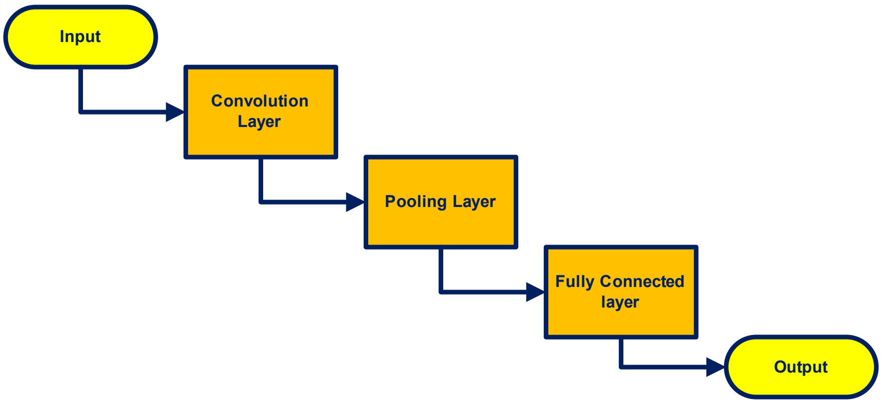
This section presents the proposed network for predicting the open price of the PETR3 stocks, as well as the metrics used.

2.1 CNN network

The convolutional neural network (CNN) performs well in digital image and natural language processing (Zhao, 2017). The CNN network can also be effectively applied to time series forecasting. The local perception and weight sharing of CNN can significantly reduce the number of parameters, thereby improving the efficiency of learning models (Li et al., 2020; Lawal et al., 2021; Santos, 2022).

CNN networks are composed of the input layer and three main layers: a convolutional layer, a pooling layer, and a fully connected layer (Figure 1) (Wu et al., 2021).

Figure 1. CNN structure.

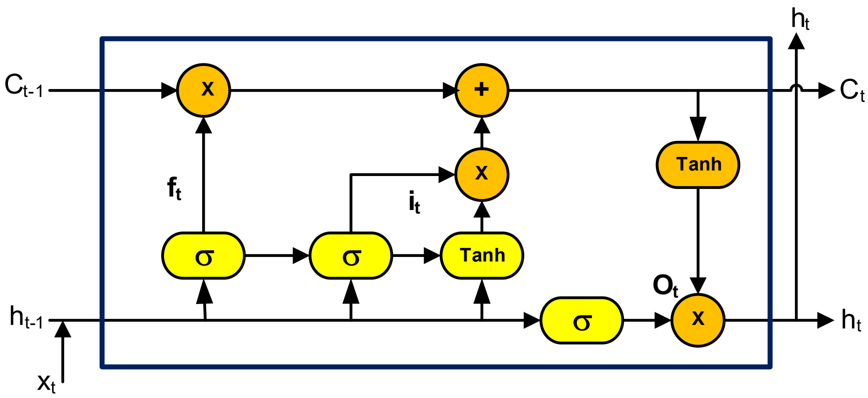


Source: The authors (2023).

2.2 LSTM network

The LSTM network can deal with long-term dependency problems and reduce the possibility of gradient disappearance (Lu et al., 2020; Santos, 2022). The topology of an LSTM network is based on memory cells (Figure 2). In the cell, the flow of information is controlled through gates (input (i_t), output (o_t), and forgetfulness (f_t)) (Santos & Chaucoski, 2020).

Figure 2. LSTM memory cell,



Source: The authors (2023).

The LSTM network equations are defined as (Lu et al., 2020; Santos, 2022; Shewalkar et al., 2019):

$$f_t = \sigma\left(w_f \cdot [h_{t-1}, x_t] + b_f\right) \quad (1)$$

$$i_t = \sigma(w_i \cdot [h_{t-1}, x_t] + b_i) \quad (2)$$

$$\tilde{C}_t = tanh(w_C \cdot [h_{t-1}, x_t] + b_C) \quad (3)$$

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t \quad (4)$$

$$o_t = \sigma(w_o \cdot [h_{t-1}, x_t] + b_o) \quad (5)$$

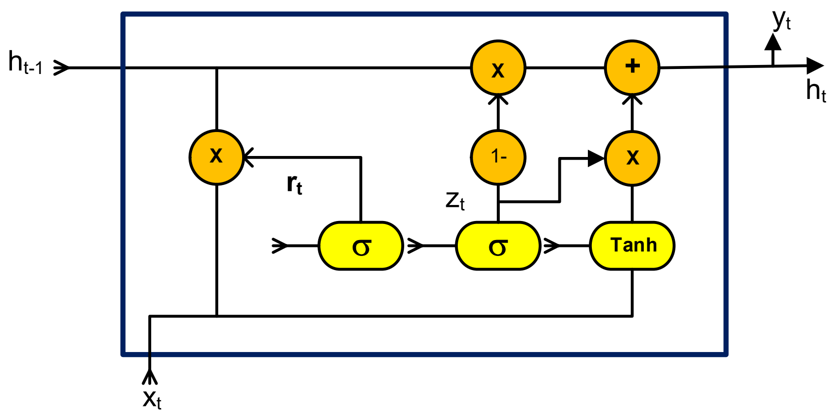
$$h_t = o_t * tanh(C_t) \quad (6)$$

Where x_t and h_t are the input and output vectors, w_f, w_i, w_c, and w_o are the weights, b_f, b_i, b_c, and b_o are the biases, σ is the sigmoid activation function, and *tanh* is the hyperbolic tangent activation function.

2.3 GRU network

The GRU network is a simplified version of the LSTM network but without performance loss. The cell of a GRU network contains two ports, a reset port z_t and an update gate r_t (Figure 3). These gates regulate the flow of input information and decide which information should be passed on to the output.

Figure 3. GRU memory cell.

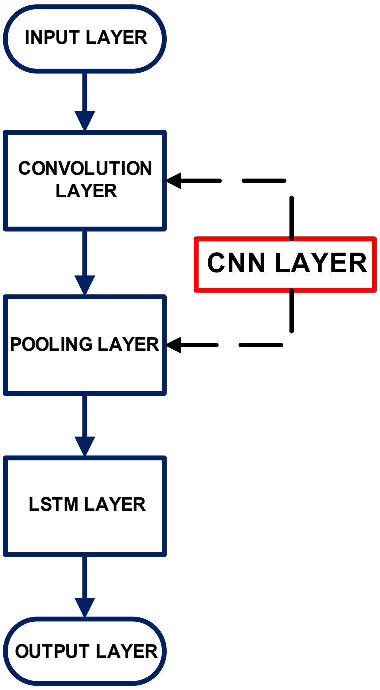


Source: The authors (2023).

2.4 CNN-LSTM network

The CNN-LSTM model has an architecture based on the CNN and LSTM networks (Figure 4). The CNN network (CNN layer) extracts features from the input data (Input Layer). In this layer, the input data passes sequentially through the convolution layer (Convolution Layer) and the pooling layer (Pooling Layer). This layer produces an output value that will be the input of the LSTM layer (LSTM Layer). The primary function of the LSTM layer is to calculate the final result of the prediction (Lu et al., 2020).

Figure 4. CNN-LSTM structure.



Source: The authors (2023).

2.5 Metrics

The MAE, RMSE, and MAPE indicators were used to evaluate the performance of the CNN-LSTM, LSTM, GRU, and CNN networks. A perfect forecast is obtained with RMSE=MAE=MAPE=0 (Pinheiro et al., 2020; Bastiani et al., 2018).

Mean Absolute Error (MAE):

$$MAE = \frac{1}{n}\sum_{1}^{n}|(y_i - \hat{y}_i)| \quad 7$$

Root Mean Square Error (RMSE):

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}((y_i - \hat{y})^2)} \quad 8$$

Mean Absolute Percentage Error (MAPE):

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}|(y_i - \hat{y}_i)/y_i| \times 100\ (\%) \qquad 9$$

Where y_i is the actual value for period i, $\hat{y}_i$ is the forecast for period i, and n is the number of observations.

3 Methodological Procedure

This section presents the methodologies used to predict the open price of PETR3 stocks.

3.1 Work steps

This work was divided into three steps:

1) Data Collection and Analysis: In the initial step, the collection and the exploratory analysis of input data from the networks were carried out.
2) Training and Validation: In this stage, the best CNN-LSTM, LSTM, GRU, and CNN models were selected.
3) Testing Stage: In the last step, the models were tested with data that did not participate in the Training and Validation stages.

3.2 Database

This work used a database obtained from the Yahoo Finance website (Yahoo Finance, 2023). The database comprises 2713 observations (Jan/2012 - Dec/2022) of the opening price of PETR3 stock of Petrobras. Each database instance has the following variables: Open, High, Low, Close, and Adj Close (Table 1).

Table 1. First five records of the data file.

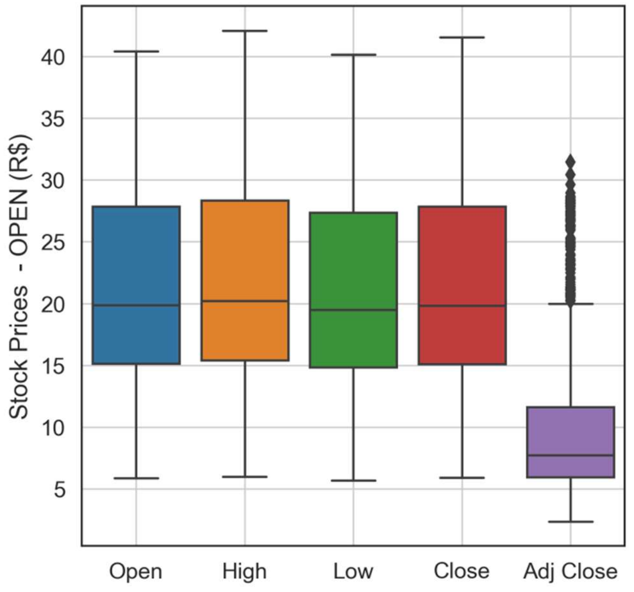
Date	Open	High	Low	Close	Adj Close
2012-01-02	23.129999	23.610001	22.740000	23.209999	8.675134
2012-01-03	23.209999	24.020000	23.209999	24.020000	9.046995
2012-01-04	23.799999	24.280001	23.570000	24.170000	9.103491
2012-01-05	24.100000	24.309999	23.760000	24.020000	9.046995
2012-01-06	24.100000	24.350000	23.910000	24.000000	9.039463

Source: The authors (2023).

Opening data (Open) refers to the asset's price when the market is opened for financial purchase and sale operations. Maximum (High) and minimum (Low) data indicate the highest and lowest value a given asset reached during financial operations on a given date. Closing data (Close) refers to the asset's price at the end of financial activities. Adjusted closing data (Adj Close) refers to the closing price after all splits and applicable dividend distributions (Pinheiro, 2022).

The box plot of the data is presented in Figure 5.

Figure 5. Box plot of stock price data.



Source: The authors (2023).

The historical series of the opening price (Open) is presented in Figure 6.

Figure 6. Opening price historical series (Open).

Source: The authors (2023).

The MinMax function was used to normalize the data. The mathematical representation of this function is presented in Equation 10 (Arunkumar et al., 2022).

$$x_n = \frac{x - x_{min}}{x_{max} - x_{min}} \quad (10)$$

Where x is data from the original series, x_{min} is the minimum value of the time series, and x_{max} is the maximum value.

4 Presentation and Analysis of Results

Initially, in this work, a descriptive analysis of the data was carried out (Table 2).

Table 2. Descriptive data analysis.

	Open	High	Low	Close	Adj Close
count	2683.000000	2683.000000	2683.000000	2683.000000	2683.000000
mean	21.015799	21.363075	20.657451	20.996802	9.148716
std	7.627991	7.707094	7.547351	7.639994	4.798853
min	5.890000	5.990000	5.670000	5.910000	2.359048
25%	15.160000	15.405000	14.855000	15.110000	5.963618
50%	19.860001	20.200001	19.500000	19.820000	7.743746
75%	27.850000	28.345000	27.380000	27.845000	11.608272
max	40.430000	42.080002	40.130001	41.560001	31.447762

Source: The authors (2023).

It can be seen, from the data presented in Table 2, that the price for Open, High, Low, Close, and Adj Close, for the period under study, averaged R$ 21.01, R$ 21.36, R$ 20.66, R$ 20.99, and R$ 9.15, respectively. With a median of R$19.86, R$20.20, R$19.50, R$19.82 and R$7.74. Also presenting, in this period, minimum prices (R$ 5.89, R$ 5.99, R$ 5.67, R$ 5.91 and R$ 2.36) and maximum (R$ 40.43, R$ 42.08, R$ 40.13, R$ 41.56 and R$ 31.45). With coefficients of variation of 36.2%, 36.4%, 36.5%, 36.3% and 52.4%. High coefficients of variation indicate data variability.

4.1Training and validation

Initially, in this work, several hyperparameters were tested. Next, the models with the best performance in the validation set were selected. These models use the Adam optimization algorithm (Adaptive moment) with the parameters presented in Table 3. The neural networks were trained with 1878 samples (70%) and validated with 805 samples (30%).

Table 3. Network parameters.

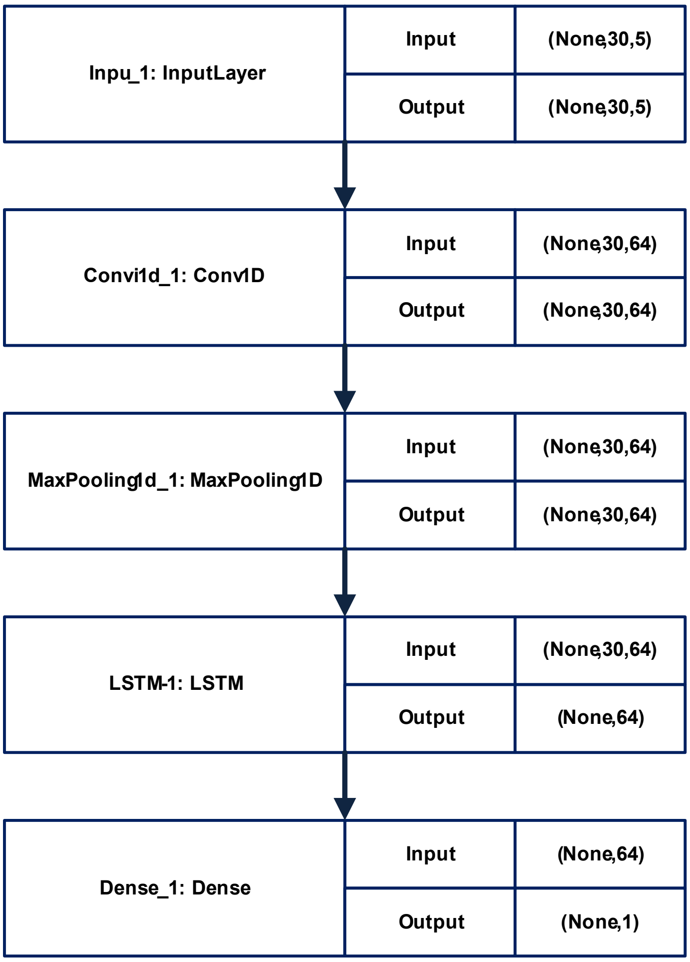
Parameters	Value
Time steps	30
Learning rate	0,001
Epochs	75
Batch	60
Convolution layer filters	64
Convolution layer kernel size	3
Convolution layer activation function	Relu
Convolution layer padding	Same

Pooling layer pool size	1
Pooling layer padding	Same
Pooling layer activation function	Relu
Numbers of hidden units in LSTM layer	64
Dropout	0.001
LSTM activation function	Relu
Dense units	1

Source: The authors (2023).

The basic structure of the CNN-LSTM model proposed in this work is presented in Figure 7. This structure was designed as follows: at the input, there is a three-dimensional data vector (None,30,5), where 30 represents the size of the time step (Time steps), and 5 is the input dimension (Open, High, Low, Close, Adj Close). Initially, the data is sent to a one-dimensional convolution layer. This layer extracts the features and produces a three-dimensional output vector (None,30,64), where 64 is the size of the Convolution layer filters. Then, the vector is sent to the pooling layer, where it is converted into the vector (None,30,64). Next, it is sent for training in the LSTM layer. The output data (None,64) goes to the dense layer, where 64 is the number of hidden units.

Figure 7. Basic Structure of CNN-LSTM.



Source: The authors (2023).

Table 4 presents the results of the metrics obtained from the models for the Validation Set.

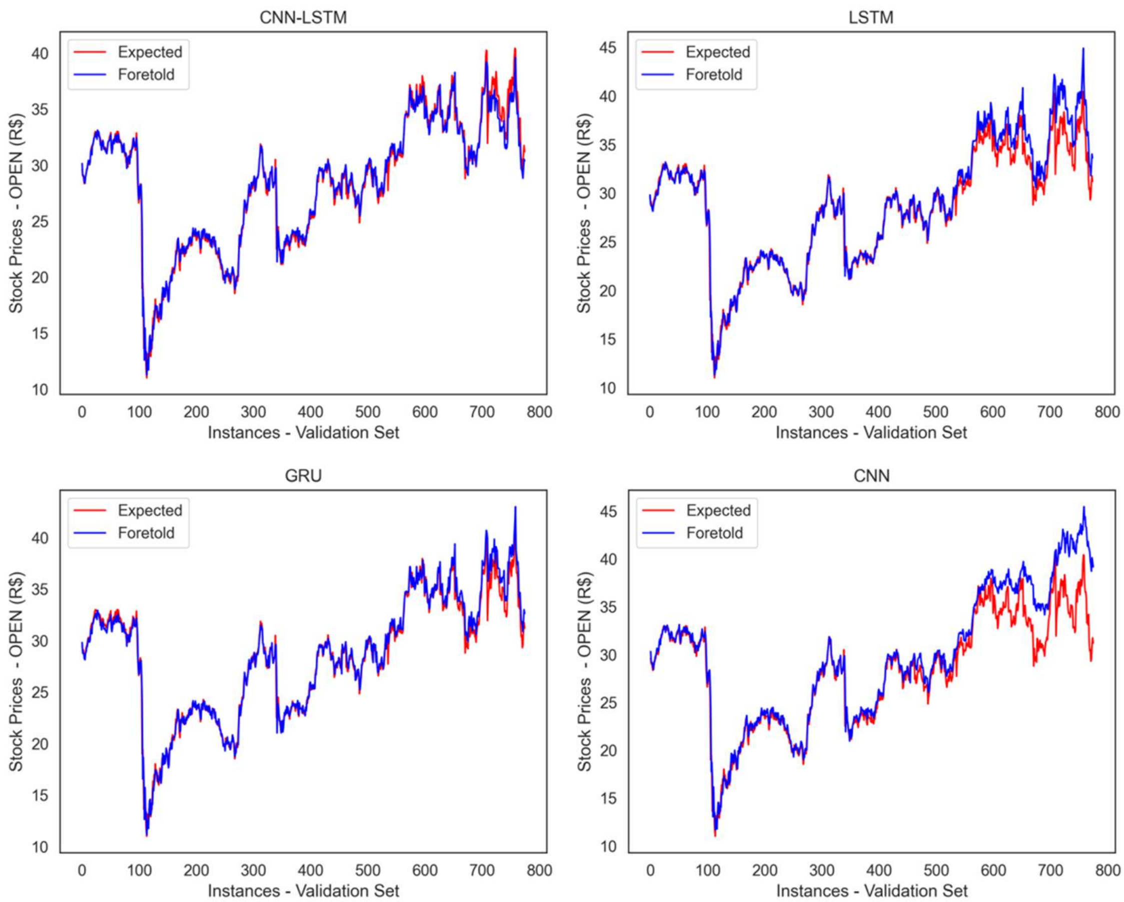
Table 4. Metrics results – Validation Set.

MODEL	MAE (R$)	RMSE (R$)	MAPE (%)
CNN-LSTM	**0,41**	**0,69**	**1,52**
LSTM	0,8	1,29	2,55
GRU	0,71	1,15	2,3
CNN	1,59	2,74	4,62

Source: The authors (2023).

The results of the metrics, MAE, RMSE, and MAPE, show that the CNN-LSTM model performed better on the validation set. It is also observed that the LSTM and GRU networks presented values for the metrics (MAE, RMSE, and MAPE) very close to the CNN-LSTM network. Figure 8 shows the prediction results of the validation set (805 days) for the four models. It can also be noted that the CNN-LSTM model achieved better adherence between the predicted data and the actual data.

Figure 8. Prediction results – Validation Set.



Source: The authors (2023).

4.2 Testing

Table 5 presents the data, observed (real) and predicted by the prediction models, for the 30 days that did not participate in the Training and Validation stage (Test Set).

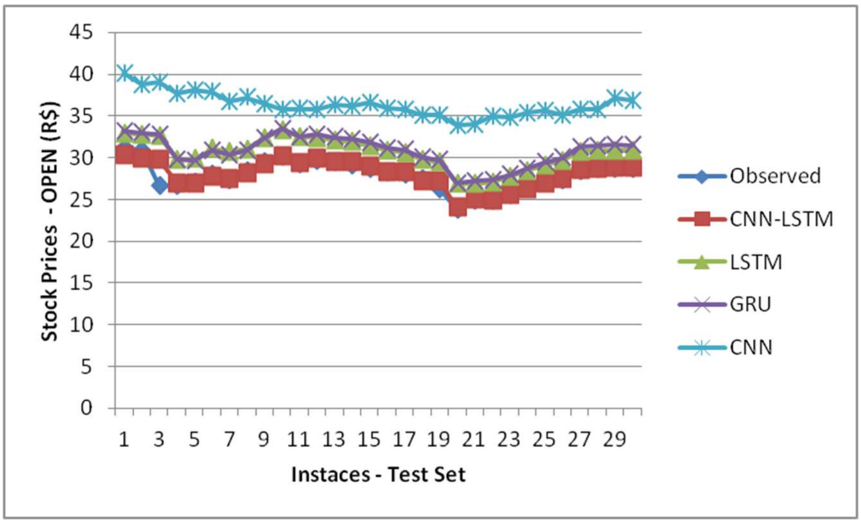
Table 5. Test Set prediction results (30 days) - (R$).

Observed	CNN-LSTM	LSTM	GRU	CNN
31,16	30,29011	32,94954	33,23226	40,10638
30,94	29,90243	32,78565	33,0044	38,78353
26,72	29,81304	32,64456	32,79764	39,00205
26,61	26,90666	29,80668	29,75706	37,63989
27,1	26,94153	29,87515	29,71688	38,10136
28,08	27,72461	31,0995	30,8478	37,89303
27,38	27,47792	30,69905	30,393	36,78247
28,39	28,09896	30,98253	30,94237	37,20356
29,55	29,19483	32,36015	32,36231	36,44544
30,22	30,20645	33,35991	33,49434	35,82314
29,24	29,44548	32,47266	32,56616	35,84384
29,7	29,87452	32,38236	32,74939	35,80054
29,5	29,54366	32,14476	32,45494	36,36739
29,17	29,45826	31,91419	32,30297	36,18704
28,74	28,97138	31,4635	31,82845	36,58517
28,24	28,32957	30,84414	31,14927	35,91081
28	28,28012	30,60405	30,9607	35,76437
27,46	27,25741	29,79876	30,05197	35,13164
26,23	27,16906	29,46479	29,72712	35,15038
23,8	24,03284	26,92745	26,89842	33,87532
24,86	25,02422	26,87778	27,16404	34,0659
24,9	24,90009	27,1145	27,33056	34,92481
25,53	25,63738	27,73749	27,96516	34,79437
26,56	26,31438	28,4824	28,73317	35,37753
27,1	26,97097	29,1325	29,46765	35,65776
27,39	27,457	29,65127	29,99907	35,15524
28,46	28,57014	30,79889	31,34539	35,76672
28,7	28,75438	30,89631	31,43199	35,74836
28,69	28,90697	30,86557	31,53908	37,1853
28,68	28,89883	30,83566	31,5014	36,92903

Source: The authors (2023).

Figure 9 presents, in graphic form, the results of the predictions for the Test Set.

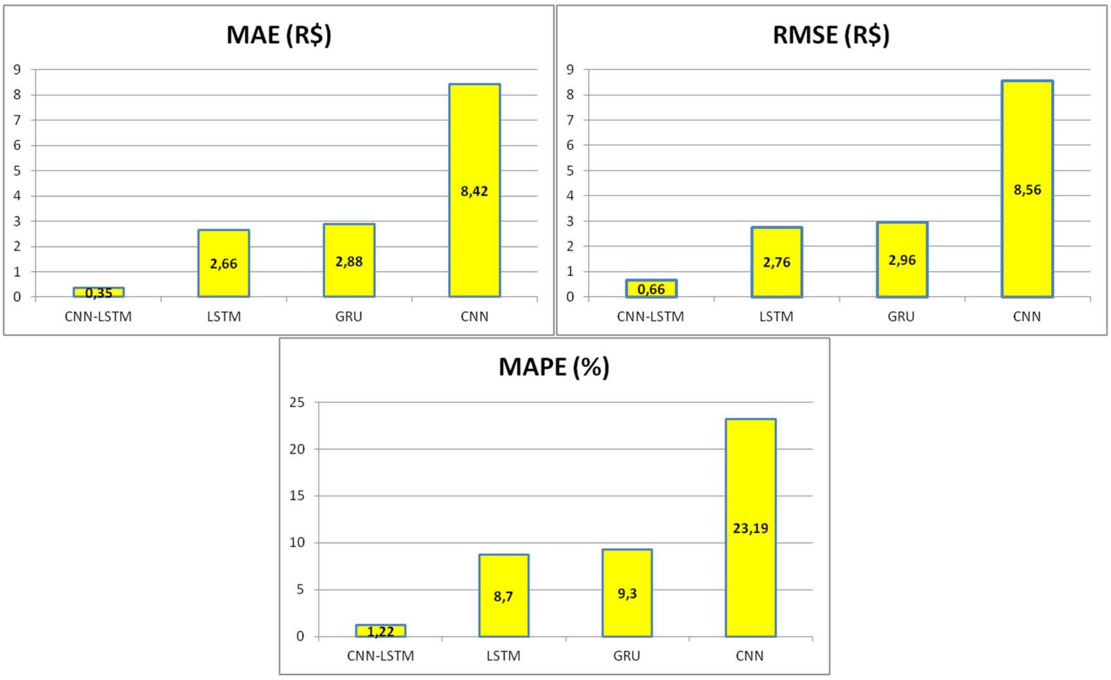
Figure 9. Predictions – Test Set



Source: The authors (2023).

Figure 10 shows the results of the MAE, RSME and MAPE metrics for the test set.

Figure 10. Results of MAE, RMSE and MAPE metrics – Test set.



Source: The authors (2023).

In Figure 9, it can be seen that the CNN-LSTM model curve better adjusted the real data curve. The CNN-LSTM model, compared with the LSTM, GRU, and CNN models, presented minor errors (MAE, RSME, and MAPE) (Figure 10). For example, the MAPE of the CNN-LSTM model, compared to the LSTM model, was reduced by 7.48%, going from 8.7% to 1.22%. Therefore, it is concluded that the CNN-LSTM model achieved greater accuracy in forecasting the prices of Petrobras PETR3 stocks. Widipruta et al. (2021) also found, when forecasting stocks in the Shanghai, Japan, Singapore, and Indonesia markets, better performance of the multivariate CNN-LSTM model. Lu et al. (2020) compared results from the CNN-LSTM network with the MLP, CNN, RNN, LSTM, and CNN-RNN networks. It concluded that the multivariate CNN-LSTM network provides a more reliable prediction for the stock price. Mahmoud and Mohammed (2020) proposed using CNN-LSTM neural networks to forecast financial time series. They concluded that combining deep learning models significantly increases the accuracy of predictions.

5 Concluding Remarks

In this study, a comparison between multivariate artificial neural network models was carried out. A daily PETR3 stock price database was used to perform this comparison. CNN-LSTM, LSTM, GRU, and CNN prediction models, implemented in the Python language, went through the training, validation, and testing stages.

Initially, the Training and the Validation of the models implemented in this work were carried out. Next, data not participating in the Training and Validation stage (Test Set) were forecasting, totaling 30 prediction days. It was observed, through the predicted data, that the CNN-LSTM model presented results close to those provided on the Yahoo Finance website. Therefore, the proximity between predicted and real values demonstrates the good generalization capacity of the CNN-LSTM model. The CNN-LSTM model presented smaller errors (MAE, RMSE, and MAPE) than the LSTM, GRU, and CNN models.

Finally, the multivariate CNN-LSTM hybrid model is valid and can help investors decide about stock market investments. A reliable forecasting model can offer insights into stock price fluctuations and provide profits to investors.

A comparison of the multivariate CNN-LSTM network with other hybrid neural network architectures is suggested for future works.

7 Referências

Althelaya, K. A., El-Alfy, E. S. M., & Mohamed, S. (2018). Stock market forecast using multivariate analysis with bidirectional and stacked (LSTM, GRU). *Saudi Computer Society*. https://doi.org/10.1109/NCG.2018.8593076

Arunkumar, K. E., Kalag, D. V., Kumar, C. M. S, Kawaji, M., & Brenza, M. (2022). Comparative analysis of Gated Recurrent Units (GRU), long Short-Term memory (LSTM) cells, autoregressive Integrated moving average (ARIMA), seasonal autoregressive Integrated moving average (SARIMA) for forecasting COVID-19 trends. *Alexandria Engineering Journal*, 31(10). https://doi.org/10.1016/j.aej.2022.01.011

Bastiani, M., Santos, J. A. A., Schmidt, C. A. P., & Sepulveda, G. P. L. (2018). Application of data mining algorithms in the management of the broiler production. *Geintec*, 8(1). https://doi.org/10.7198/geintec.v8i4.1275

Chen, K., Zhou, Y., & Dai, F. (2015). *A LSTM-based method for stock returns prediction: A case study of China stock market*. Big Data. https://doi.org/10.1109/BigData.2015.7364089

Haykin, S. (2001). *Neural networks: a comprehensive foundation*. New Delhi: Pearson Prentice Hall.

Khaidem, L., Saha, S., & Dey, S. R. (2016). Predicting the direction of stock market prices using random forest. *Applied Mathematical Finance*. https://doi.org/10.48550/arXiv.1605.00003

Lawal, A., Rehna, S., Alems, L. M., & Alam, M. (2021). Wind speed prediction using hybrid 1D CNN and BLSTM network. *IEEE Acess*, 9(1). https://doi.org/10.1109/ACCESS.2021.3129883

Li, P., Abdel-aty, M., & Yuan, J. (2020). Real-time crash risk prediction on arterials based on LSTM-CNN. *Accident Analysis and Prevention*, 135(1). https://doi.org/10.1016/j.aap.2019.105371

Lu, W.; Li, J., Wang, J., & Qin, L. (2020). A CNN-BiLSTM-AM method for stock price prediction. *Neural Compunting and Applications*, 33(1). https://doi.org/10.1007/s00521-020-05532-z

Mahmoud, A., & Mohamed, (2020). A. A Survey on Deep Learning for Time-Series Forecasting. *Machine Learning and Big Data Analytics Paradigms: Analysis, Applications and Challenges*. 77(1). https://doi.org/10.1007/978-3-030- 59338-4_1637

Neyra, J. R., & Moura, T. R. S. (2022). Initial study of application of Neural Networks to predict flow patterns in multiphase flow. *Scientia Plena*. 18(1). https://doi.org/10.14808/sci.plena.2022.084802

Pinheiro, T. C., Santos, J. A., & Pasa, L. A. (2020). Gestão da produção de frangos de corte por meio de redes neurais artificiais. *Revista Holos*, 2(1). https://doi.org/10.15628/holos.2020.9043

Pinheiro, T. J. (2022). *Modelos de deep learning e previsão de preço de ações: estudo de casos da bolsa brasileira*. Avaliable at: https://tede2.uepg.br/jspui/bitstream/prefix/3672/1/Thiago%20Jos%c3%a9%20Pinheiro.pdf

Samarawickrama, A., & Fernando, T. (2017). A recurrent neural network approach in predicting daily stock prices an application to the Sri Lankan stock market. *Conference on Industrial and Information Systems*. https://doi.org/10.1109/ICIINFS.2017.8300345

Santos, J. A. A. (2022). Previsão do preço do café arábica: uma aplicação de redes neurais CNN-BLSTM. *Research, Society and Development*, 8(2). https://doi.org/10.33448/rsd-v11i3.26101

Santos, J. A. A.; Chaukoski, Y. (2020). Previsão do consumo de energia elétrica na região sudeste: um estudo de caso usando SARIMA e LSTM. *Revista Cereus*, 12(4). https://doi.org/10.18605/2175-7275/cereus.v12n4p93-104

Sebastian, S. (2016). Performance evaluation by artificial neural network using WEKA. *International Research Journal of Engineering and Technology*, 3(3), 2016.

Shewalkar, A., Nyavanandi, D., & Ludiwig, S. (2019). A. Performance evaluation of deep neural networks applied to speech recognition; RNN, LSTM and GRU. *JAICR*, 9(4). https://doi.org/10.2478/jaiscr-2019-0006

Song, H.; & Choi, H. (2023). Forecasting Stock Market Indices Using the Recurrent Neural Network Based Hybrid Models: CNN-LSTM, GRU-CNN, and Ensemble Models. *Applied Sciences*. 13(1). https://doi.org/10.3390/app13074644

Widepruta, H.; Mailangkay, A., & Gautama, E. (2021). Multivariate CNN-LSTM model for p Financial Time-series prediction. *Hindawi Complexity*, 1(1). https://doi.org/10.1155/2021/9903518

Wu, K., Wu, J., & Feng, L., Yang, B.; Liang, R., & Yang, S. (2021). An attention-based CNN-LSTM-BiLSTM model for short-term electric load forecasting in integrated energy system. *International Transactions on Electrical Energy Systems*, 31(1). https://doi.org/10.1002/2050-7038.12637

Yahoo Finance. (2023). *Petróleo Brasileiro S.A. - Petrobras (PETR3.SA)*. Avaliable at: https://finance.yahoo.com/quote/PETR3.SA/history?p=PETR3.SA

Yan, X., Weihan, W., & Chang, M. (2020). Research on financial assets transaction prediction model based on LSTM neural network. *Neural Computing and Applications*, 33(1). https://doi.org/10.1007/s00521-020-04992-7

Zha, W., Liu, Y., Wan, Y., Luo, R., Li, D., Yang, S., & Xu, Y. (2022). Forecasting monthly gas field production based on the CNN-LSTM model. *Energy*, 260(1). doi: 10.1016/j.energy.2022.124889 637

Zhao, Z. (2017). LSTM network: a deep learning approach for short-term traffic forecast. *IET Intelligent Transport Systems*, 11(2). https://doi.org/10.1049/iet-its.2016.0208