Prediction of Future Stock Price Using Recurrent Neural Network

Nur Izzah Atirah Mohd Ikhram[1], Nor Hayati Shafii[2]*, Nur Fatihah Fauzi[3], Diana Sirmayunie Md Nasir[4], Nor Azriani Mohd Nor[5]
[2,3,4,5]College of Computing, Informatics, and Mathematics, Universiti Teknologi MARA Perlis Branch, Arau Campus, 02600 Arau, Perlis, Malaysia.

*Corresponding author: *norhayatishafii@uitm.edu.my*

Received Date: 20 February 2023
Accepted Date: 16 April 2023
Revised Date: 5 May 2023
Published Date: 1 September 2023

HIGHLIGHTS

- A machine learning process is applied to predict the stock price, which is very volatile, dynamic, and non-linear.
- The recurrent neural network algorithm is utilized to develop an adequate model for prediction.
- Rapid Miner software is used to establish the recurrent neural network's structure and the model's accuracy is measured based on the mean square error and root mean square error values.

ABSTRACT

The stock market can affect businesses in various ways, as the rise and fall of a company's share price values impact its market capitalization and overall market value. However, forecasting stock market returns is challenging because financial stock markets are unpredictable and non-linear, with factors such as market trends, supply and demand ratios, global economies, and public opinion affecting stock prices. With the advent of artificial intelligence and increased processing power, intelligent prediction techniques have become more effective in forecasting stock values. This study proposes a Recurrent Neural Network (RNN) model that uses a deep learning machine to predict stock prices. The process includes five stages: data analysis, dataset preparation, network design, network training, and network testing. The accuracy of the model is determined by the mean square error (MSE) and root mean square error (RMSE), which are 1.24 and 1.12, respectively. The predicted closing price is then compared to the actual closing price to assess the accuracy of the model. Finally, it is suggested that this approach can also be used to forecast other volatile time-series data.

Keywords: *Stock Price, Prediction, Recurrent Neural Network, Rapid Miner*

INTRODUCTION

The stock price is an important metric for measuring the value of a company, and it represents the current market value of a share of its stock. This value is determined by the supply and demand of the stock in the market (Pinsent 2021). As such, the stock price can fluctuate frequently due to a variety of factors, including global events, economic conditions, and company performance. Investors pay close attention to stock prices as they are a reflection of a company's market capitalization, which is the total value of all outstanding shares of a company's stock (Kramer 2021). An increase or decrease in a stock's price directly affects the market capitalization, making stock prices a crucial consideration for investors in making investment decisions.

Stock price forecasting is a challenging task due to several factors such as stock price volatility, dynamism, and nonlinearity. It becomes even more challenging due to the influence of macro and micro factors such as political events, global economic circumstances, unforeseen occurrences, and a company's financial performance. As Li (2021) pointed out, accurately predicting stock prices is extremely difficult due to these factors. Some investors may also have different strategies when it comes to buying or selling stocks. For example, according to Yates (2021), some investors may avoid buying stocks or indexes that have risen rapidly as they believe a correction is due, while others may avoid selling stocks because they anticipate further declines. These varying investment strategies can also contribute to the difficulty of accurately forecasting stock prices.

Stock prediction is defined as the act of attempting to forecast the future value of a company’s stock or other financial instruments traded on an exchange (*Stock Market Prediction Definition*, 2018). Prediction approaches can be divided into three groups that overlap: fundamental analysis, technical analysis, and technology procedures. Fundamental analysis is the study of economic factors that affect the price of a stock. Meanwhile, the technical analysis is used to predict what other stockholders are thinking based on available information about stock price and volume. Stock market forecasting has moved into the technological arena with the invention of the digital computer.

According to Abu-Mustofa and Atiya (1996), forecasting the stock market is extremely difficult due to its dynamic and non-linear nature. Neural networks are among those capable of stimulating nonlinear market behavior. Some of the most recent stock market prediction tools include Artificial Neural Networks, Neuro-Fuzzy Systems, Time Series Linear Models (TSLM), and Recurrent Neural Networks (RNN) (Polamuri et al., 2020). Because there are so many different kinds of neural networks, choosing the right one is critical because it can have a big impact on forecast accuracy.

Recurrent neural network (RNN) is a type of advanced artificial neural network (ANN) that uses memory-directed cycles. RNN is a sort of neural network that includes internal memory functions via feedback loops and could save data while processing fresh inputs. This type of memory makes it excellent for operations that need consideration of historical inputs (Zheng & He, 2021).

RNNs are also frequently used to address problems with sequential input data, such as time series. RNNs are a type of neural network that remembers what it has previously processed and can thus learn from previous iterations during training. RNNs can process variable-length input sequences by utilizing their internal state (memory). Because RNNs are potentially turing complete, they can execute arbitrary input sequences and run arbitrary programming (Onnen, 2021). The term "turing complete" refers to the ability to calculate anything that any other computing technique can compute.

Techopedia (2018) stated that the capacity to build on previous types of networks with fixed-size input and output vectors is one feature of RNNs. In an RNN, node connections create a directed graph along a sequence, allowing it to display dynamic temporal behavior for a time sequence (Jahan, 2018). According to Shriram et al. (2021), RNN is recurrent because it performs the same function for each data input and the outcome of the current calculation is dependent on the previous calculation. The generated output is replicated and returned to the recurrent network. It decides by evaluating the current input and the output that it has learned from the previous input.

The ANN type of network has no memory of the information it receives and is a poor predictor. Meanwhile, the information in RNN is looping. When making a decision, it considers the current input as well as what it has learned from previous inputs (Donges, 2021). RNNs, unlike feed-forward neural networks, may process input sequences using their internal state (memory). In other neural networks, all of the inputs are unrelated to one another. However, with RNN, all of the inputs are connected (Shriram, et al., 2021).

METHODOLOGY

In this study, a sample dataset was obtained from Malaysian Pacific Industries' (MPI) webpage on the Investing.com website, which showed the daily stock price over three years, from 2 January 2019 to 31 December 2021, comprising 734 data points. This sample size is expected to provide valuable information and improve the accuracy of the model. The data was analyzed and the stock price was forecasted using RapidMiner Server 9 software. RapidMiner is a visual workflow designer tool that helps speed and automate the creation of visual models mainly for non-coding domain experts. The recurrent neural network (RNN) model is created using a sequential model. According to Kurnaz & Demir (2021), the sequential model is one of the most basic types of models, consisting of a stack of layers. These layers can be thought of as the basic building elements of the neural network.

Each improved and tested model was used to forecast the trends. The performance of the suggested models was assessed using Mean Square Error (MSE) and Root Mean Square Error (RMSE).

$$MSE = \frac{\sum_{i=1}^{N} (y_i - F_i)^2}{N} \tag{1}$$

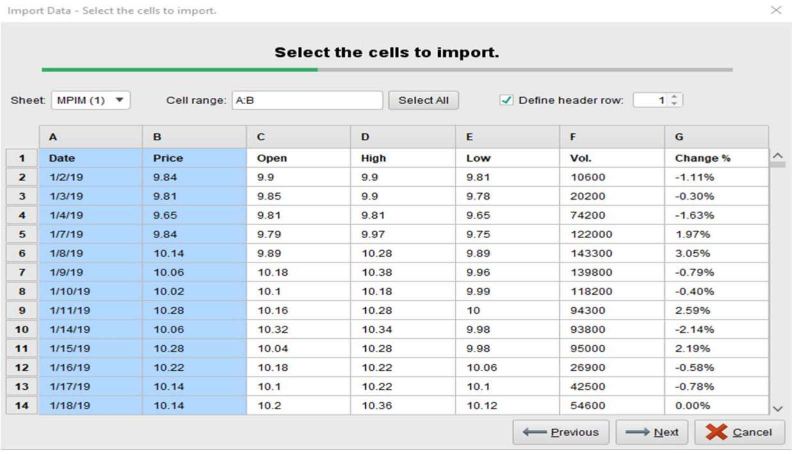
$$RMSE = \sqrt{\frac{\sum_{i=1}^{N} (y_i - F_i)^2}{N}} \tag{2}$$

y_i is the actual values
F_i is the predicted values
N is the number of the data

MSE and RMSE are used to evaluate the accuracy of forecasted values. RMSE measures the difference between the actual and predicted data, providing insight into the extent of the differences between the predicted and actual results.

Step 1: Data Collection

To import the dataset into RapidMiner, we first saved it as a Microsoft Excel 97-2003 Worksheet as shown in Figure 1. Then, we opened RapidMiner and loaded the dataset by selecting the appropriate file. After loading the data, we selected the date and price columns by using the 'Select Attributes' operator and deselected all other columns. This allowed us to focus our analysis on the two most important variables for forecasting stock prices.



Import Data - Select the cells to import.

Select the cells to import.

Sheet: MPIM (1) ▼ Cell range: A:B [Select All] ☑ Define header row: 1

	A	B	C	D	E	F	G
1	Date	Price	Open	High	Low	Vol.	Change %
2	1/2/19	9.84	9.9	9.9	9.81	10600	-1.11%
3	1/3/19	9.81	9.85	9.9	9.78	20200	-0.30%
4	1/4/19	9.65	9.81	9.81	9.65	74200	-1.63%
5	1/7/19	9.84	9.79	9.97	9.75	122000	1.97%
6	1/8/19	10.14	9.89	10.28	9.89	143300	3.05%
7	1/9/19	10.06	10.18	10.38	9.96	139800	-0.79%
8	1/10/19	10.02	10.1	10.18	9.99	118200	-0.40%
9	1/11/19	10.28	10.16	10.28	10	94300	2.59%
10	1/14/19	10.06	10.32	10.34	9.98	93800	-2.14%
11	1/15/19	10.28	10.04	10.28	9.98	95000	2.19%
12	1/16/19	10.22	10.18	10.22	10.06	26900	-0.58%
13	1/17/19	10.14	10.1	10.22	10.1	42500	-0.78%
14	1/18/19	10.14	10.2	10.36	10.12	54600	0.00%

Figure 1: Import the dataset into RapidMiner

The price column's function has been changed to label. The price is the target to predict the value for the future process.

Step 2: Data Preparation

The dataset was divided into two parts using a commonly used 70:30 ratio, where 70% (514) of the data was allocated for training and the remaining 30% (220) was reserved for testing as shown in Table 1. The training set was used to train the model, while the testing set was used to assess the performance of the classifier. The split operator was employed during the partitioning process of the data.

Table 1: Data partition

Data partition	Total data	Date
Training set	514	2nd January 2019 – 9th February 2021
Testing set	220	10th February 2021 – 31st December 2021
Validation	734	2nd January 2019 – 31st December 2021

The same testing set data was used in the validation process, which involved retrieving the MPI data into the process as in Figure 2. The cross-validation operator was employed to validate the data, using two sub-processes: training and testing. The model was trained using the training sub-process, and then the trained model was used in the testing sub-process. During the testing phase, the model's performance was assessed as presented in Figure 3.

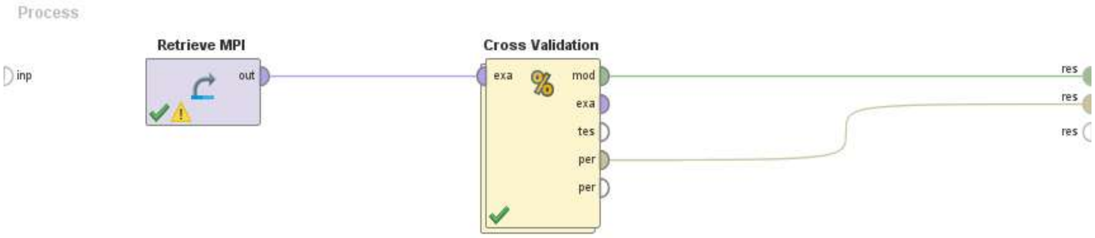


Figure 2: Structure of the validation process

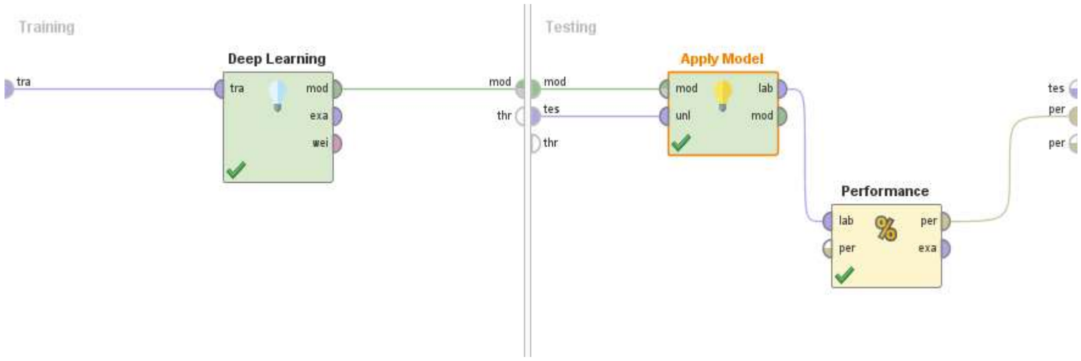


Figure 3: The sub-process model for the validation process

Step 3: Network Design

During this phase, the structure of the neural network is determined, with the deep learning operators playing a crucial role in the process. Typically, a basic recurrent neural network consists of three layers, including an input layer, a hidden layer, and an output layer as shown in Figure 4. In this model, there is only one node for the input and output layers, while two hidden layers are utilized. After setting up the layers, the apply model operator is then employed to the data to apply deep learning.

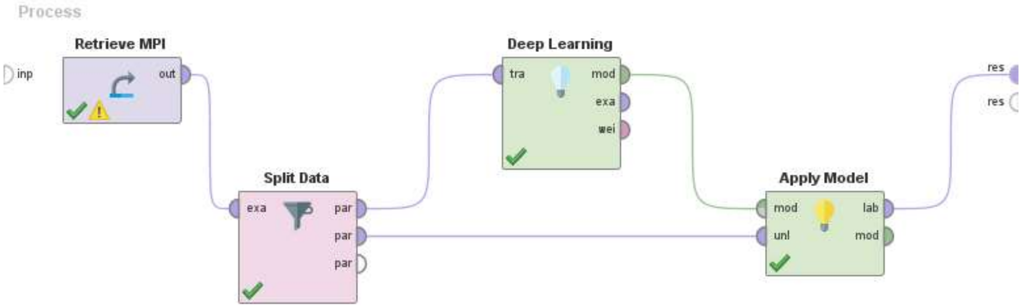


Figure 4: Structure of RNN model in RapidMiner Software

Step 4: Network Training

The model was then trained over ten epochs with different sample values. Because the data is too large, 10 epochs are chosen to allow the data to pass the algorithm in smaller batches. The number of epochs is critical in determining the model that best describes the sample data with the least amount of inaccuracy.

The algorithm used to train the network is a linear regression algorithm that plots a line based on the dependent and explanatory variables. For each epoch, the training was based on root mean square error

(RMSE), deviance, mean absolute error (MAE), and R-squared(R^2). The results for the epochs training are shown in Table 2.

Table 2: Scoring history

Epochs	Iterations	Samples	Training RMSE	Training Deviance	Training MAE	Training R^2
1.00000	1	440.00000	5.52831	30.56226	4.66010	0.85030
2.00000	2	880.00000	4.77423	22.79331	4.14316	0.88835
3.00000	3	1320.00000	3.55286	12.62279	3.03413	0.93817
4.00000	4	1760.00000	3.27209	10.70658	2.78926	0.94756
5.00000	5	2200.00000	1.97516	3.90125	1.52041	0.98089
6.00000	6	2640.00000	1.53726	2.36317	1.20370	0.98842
7.00000	7	3080.00000	1.33786	1.78986	0.99880	0.99123
8.00000	8	3520.00000	1.76843	3.12734	1.46408	0.98468
9.00000	9	3960.00000	1.11574	1.24488	0.83433	0.99390
10.00000	10	4400.00000	1.88120	3.53891	1.69997	0.98267
10.00000	10	4400.00000	1.11574	1.24488	0.83433	0.99390

The data was trained using error measures such as RMSE, Deviance, MAE, and R^2, which were used to determine the optimal model value and to measure the accuracy and goodness-of-fit of the forecasted values. A lower RMSE and MAE indicate a closer fit to the actual data, while a high R^2 value close to one indicates a good correlation between predicted and actual values. Based on the values presented in Table 2, the error measures increased after nine iterations, suggesting that further iteration was not significant. In conclusion, the best network model was achieved after nine iterations with the lowest RMSE (1.11574), lowest MAE (0.83433), and an R^2 value of 0.9939, which was very close to one.

Step 5: Network Testing

In the final phase, the testing set is used to evaluate the performance of the network model. To assess the performance of the model, the performance operator is added to the model process.

FINDINGS AND DISCUSSION

Figure 5 shows a graph of actual and forecasted stock prices over 220 days. Despite the sharp decline in the forecasted data, the actual stock price of MPI does not exhibit a significant difference from the forecasted stock price, as indicated by the graph.

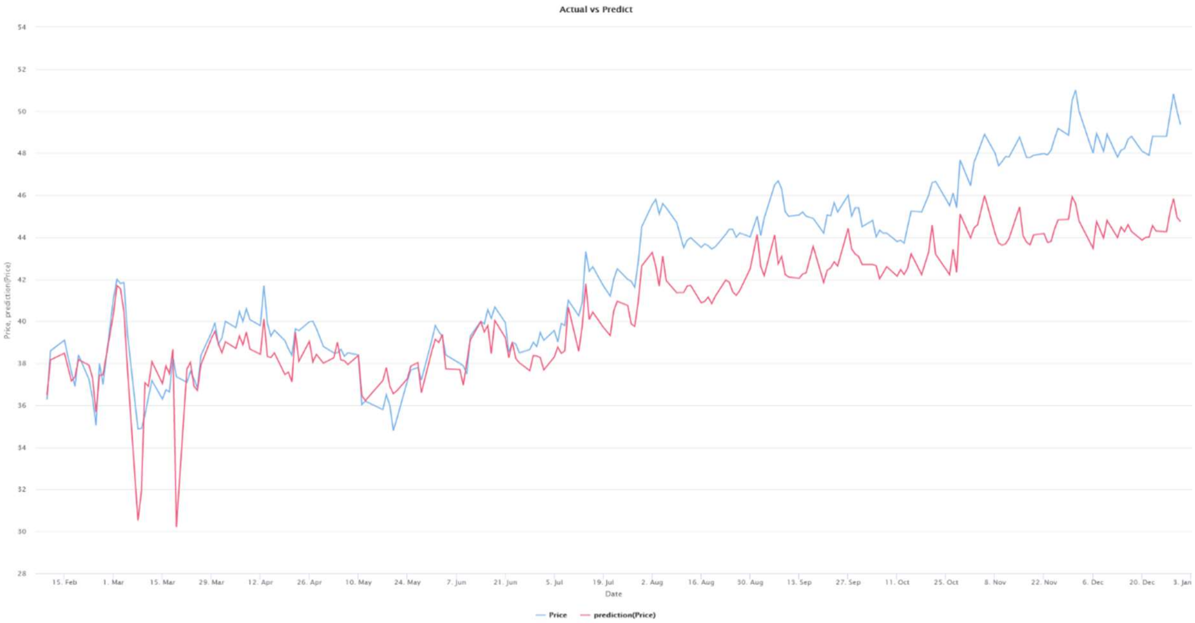


Figure 5: Graph of actual and predicted price

From the measurement score of MSE and RMSE, we can conclude that the forecasts were reasonably accurate.

Table 3: Summary of the result

Data	MSE	RMSE
Target	1.3604233	1.1663718
Output	1.2448792	1.1157416
R^2 = 0.9939022 MAE = 0.8343272		

Based on the testing set results presented in Table 3, it can be observed that the target value for MSE is 1.3604233, while the model output is 1.2448792. Similarly, the target and output values for RMSE are 1.1663718 and 1.1157416, respectively. A smaller output value indicates that the model is more accurate. From the experimental results, it can be concluded that RNN is an appropriate method for time series forecasting as it has produced low MSE, RMSE, and mean absolute error (MAE) values.

CONCLUSION AND RECOMMENDATIONS

This study proposed a Recurrent Neural Network (RNN) as a potential solution for predicting stock prices, as RNN is designed to handle time-based problems and can predict future values. The feasibility and performance of RNN were thoroughly investigated in this study, and the findings suggest that it may be one of the best approaches for accurately forecasting closing prices in the stock market. The experimental results showed a small difference of only 8.49 percent between the targeted and output MSE, indicating that the RNN model can effectively identify patterns and accurately predict stock prices even in volatile market conditions, with a high level of 95.66 percent accuracy. While RNN is effective for predicting stock prices, it's important to remember that the outcomes may vary depending on factors such as the type of data, the current market situation, and the skill and experience of the forecaster. Additionally, the stock market can be highly unpredictable, and changes in stock prices may not follow consistent patterns, leading to variations in actuality and accuracy. As with any forecasting method, it's important to use multiple models and techniques and continually evaluate their performance.

Several suggestions and ideas for future research could further improve the accuracy of stock price prediction. Firstly, the data should be updated in real time using the same method to ensure the most recent information is incorporated into the prediction model. Secondly, it is recommended to develop deep learning models that can simultaneously analyse and incorporate the latest data to anticipate market changes. Utilizing more data will likely lead to stronger and more reliable models, resulting in a lower margin of logic error in forecasts. Additionally, multiple tests should be conducted on the neural network to determine the optimal number of epochs and hidden layers, which could be further improved by developing a mathematical formula to determine the ideal neural network architecture. Furthermore, other types of deep learning models, such as Convolutional Neural Networks (CNN) and Long Short-Term Memory Networks (LSTM), could be tested for comparison to the RNN model. The use of additional data sources, such as news articles, social media sentiment analysis, and economic indicators, could potentially improve the accuracy of the model. Finally, the effectiveness of the proposed model could be tested in different markets and with different companies to verify its generalizability. These suggestions could help improve the accuracy and reliability of the model for real-world applications.

CONFLICT OF INTEREST DISCLOSURE

The authors declared that they have no conflicts of interest to disclose.

REFERENCES

Abu-Mustofa, Y. S. & Atiya, A. F. (1996). Introduction to Financial Forecasting. Applied Intelligence, 6(3).

Donges, N. (2021). A Guide to RNN: Understanding Recurrent Neural Networks and LSTM Networks. Built In. https://builtin.com/data-science/recurrent-neural-networks-and-lstm

Jahan, I. (2018). Stock Price Prediction Using Recurrent Neural Networks. https://library.ndsu.edu/ir/bitstream/handle/10365/28797/Stock%20Price%20Prediction%20Using%20Recurrent%20Neural%20Networks.pdf?sequence=1&isAllowed=y

Kramer, L. (2021). How Is a Company's Stock Price and Market Capitalization Determined. Investopedia. https://www.investopedia.com/ask/answers/how-companys-stock-price-and-market-cap-determined/

Kurnaz, G. & Demir, A. S. (2021). Prediction of SO2 and PM10 Air Pollutants Using a Deep Learning-Based Recurrent Neural Network: Case of Industrial City Sakarya. Urban Climate, 41 (101051).

Li, K. (2021). Predicting Stock Price Using Machine Learning. Neptune Blog. https://neptune.ai/blog/predicting-stock-prices-using-machine-learning

Onnen, H. (2021). Temporal Loops: Intro to Recurrent Neural Networks for Time Series Forecasting in Python. Towards Data Science. https://towardsdatascience.com/temporal-loops-intro-to-recurrent-neural-networks-for-time-series-forecasting-in-python-b0398963dc1f

Pinsent, W. (2021). Understanding Stock Prices and Values. Investopedia. https://www.investopedia.com/articles/stocks/08/stock-prices-fool.asp

Shriram, S., Anuradha, K., & Uma, K. P. (2021). Future Stock Price Prediction using Recurrent Neural Network, LSTM and Machine Learning. ICRADL, 09(05), 304-308.

Stock market prediction definition. (2018). Capital.Com. https://capital.com/stockmarket-prediction-definition

Yates, T. (2021). 4 Ways to Predict Market Performance. Investopedia. https://www.investopedia.com/articles/07/mean_reversion_martingale.asp

Techopedia. (2018). Recurrent Neural Network (RNN). https://www.techopedia.com/definition/32834/recurrent-neural-network-rnn

Zheng, L. & He, H. (2021). Share Price of Aerospace Relevant Companies with Recurrent Neural Networks Based on PCA. Expert System *with Application*, 183(115384).