Applied Computer Science, vol. 19, no. 3, pp. 160–173
doi: 10.35784/acs-2023-30

Submitted: 2023-06-03 | Revised: 2023-08-10 | Accepted: 2023-09-10

Keywords: Stock Prediction, Deep Learning, GRU, EMA, ReLU, AntiReLU

Erizal ERIZAL [0000-0002-3937-3483]*, *Mohammad DIQI* [0000-0002-9012-9080]**

PERFORMANCE EVALUATION OF STOCK PRICE PREDICTION MODELS USING EMAGRU

Abstract

Stock price prediction is an exciting issue and is very much needed by investors and business people to develop their assets. The main difficulties in predicting stock prices are dynamic movements, high volatility, and noises caused by company performance and external influences. The traditional method investors use is the technical analysis based on statistics, valuation of previous stock portfolios, and news from the mass media and social media. Deep learning can predict stock price movements more accurately than traditional methods. As a solution to the issue of stock prediction, the authors offer the Exponential Moving Average Gated Recurrent Unit (EMAGRU) model and demonstrate its utility. The EMAGRU architecture contains two stacked GRUs arranged in parallel. The inputs and outputs are the EMA10 and EMA20, formed from the closing prices over ten years. The authors also combine the AntiReLU and ReLU activation functions into the model so that EMAGRU has 6 model variants. The proposed model produces low losses and high accuracy. RMSE, MEPA, MAE, and R^2 are 0.0060, 0.0064, 0.0050, and 0.9976 for EMA10, and 0.0050, 0.0058, 0.0045, and 0.9982 for EMA20, respectively.

1. INTRODUCTION

Noise and volatility are the main challenges in predicting the stock market (Jin et al., 2020). Investors and businesses strive to maximize profits and reduce risk by utilizing stock predictions, primarily through technical and fundamental analysis (Nti et al., 2020). Both studies use historical transaction data (Zhao et al., 2021) and stock performance portfolios (Chen, Zhang et al., 2021). This approach is obsolete in today's technological and market-based society (Gao et al., 2021).

Investors and the public require accurate stock predictions to increase returns (Chun et al., 2021) and build broader business insights (Jabeen et al., 2021). Ensuring a significant confidence level in stock trends is not easy (Chen, Jiang, et al., 2021) because stock price movements rest not only on the company's performance but also on social situations, monetary dynamics, and other external influences (Khan et al., 2022). The changes in stock prices that are formed every day come from a combination of various unsolved variables (Li et al., 2020).

* Universitas Respati Yogyakarta, Faculty of Science and Technology, Department of Information System, Indonesia, erizal@respati.ac.id
** Universitas Respati Yogyakarta, Faculty of Science and Technology, Department of Informatics, Indonesia, diqi@respati.ac.id

Several studies were the forerunners in applying machine learning to the challenge of stock prediction. RNN could remember the historical context in stock predictions (Saud & Shakya, 2021). Nabipour et al. (2020) examined ten widely used technical indicators with several machine learning methods to determine which was superior for predicting stock prices. Compared to Adaboost, Gradient Boosting, XGBoost, and Decision Tree, LSTM produced the best and most accurate model fittings. Liwei et al. (2021) and Lv et al. (2021) believed that adding Bayesian calculations could optimize the XGBoost-LSTM and LightGBM-LSTM models. In addition, the combination of ARIMA (Chiniforoush & Latif Shabgahi, 2021) and LSTM provided better results for short-term predictions (Wang et al., 2021).

Ta et al. (2020) researched to anticipate market movements with the LSTM network and optimize portfolio performance. The proposed LSTM prediction model provided better accuracy than Linear Regression and SVM. LSTM with LASSO provided better predictive capabilities than PCA (Gao et al., 2021).

Thormann et al. (2021) combined LSTM with Twitter's financial and sentiment features to predict stock value, which outperforms the baseline in all situations. LSTM-based stock price sentiment analysis improved RMSE (Ko & Chang, 2021; Zhang & Fang, 2021). Ji et al. (2021) proposed Deep Learning-based stock predictions that combine LSTM with Doc2Vec, SAE, and wavelet transformations. The Doc-W-LSTM model gave a coefficient of determination of R^2 = 0.957. Qiu et al. (2020) proposed WLSTM+Att and showed RMSE 0.1971, MAE 0.1569 and R^2 0.9621.

Some researchers proposed several model algorithms other than LSTM. Lin et al. proposed a stock prediction model using the Generative Adversarial Network (GAN). The model generator used GRU, and the discriminator used CNN. The performance of this model was better than that of the standard model (Lin et al., 2021). Diqi et al. (2022) proposed a GAN model and gave R^2 real = 0.811166, and R^2 synthetic = 0.674971.

Shahi et al. (2020) compared LSTM and GRU to include financial news sentiment. This model yielded R^2 = 0.967 for GRU, and R^2 = 0.979 for LSTM. Diqi (2022) used LSTM to predict EMA10 and EMA20 and obtained RMSE 0.00714 and MAPE 0.07705 for EMA10 and RMSE 0.00355 and MAPE 0.05273 for EMA20. Lu et al. (2020) combined LSTM with MLP, CNN, and RNN. CNN-LSTM had a higher and better prediction accuracy for stock price predictions, i.e. R^2 = 0.9646 (Diqi et al., 2023). The accuracy of stock predictions on LSTM models could be improved by integrating GRU and RNN (Lv et al., 2021).

Manjunath et al. (2021) and Savadi Hosseini and Ghaderi (2020) analyzed the model results using RNN, LSTM, and three GRU variations. The GRU variant outperformed LSTM and RNN. Saud and Shakya (2021) suggested that the 3-GRU model predicted the next day's closing price after comparing it with GRU-MACD, GRU-KST, GRU-ADX, and GRU-ALL. Radojičić and Kredatus (2020) used Fourier Transforms to extract new features and offered statistically significant improvements in GRU model performance.

This research addressed the challenges of predicting stock prices, including dynamic movements, high volatility, and noises caused by company performance and external influences. We proposed a new model called Exponential Moving Average Gated Recurrent Unit (EMAGRU) to predict stock prices accurately and reliably. The EMAGRU model consists of two layers of parallel Gated Recurrent Units (GRUs) with ReLU and AntiReLU activation functions. The exponential moving average (EMA) of the last 50-day closing price is used as input for the EMAGRU model. We constructed six EMAGRU models based on this concept. These models allow us to predict stock prices for 20 trading days. According to the findings, it is possible to attain maximal R2 while simultaneously minimizing RMSE, MAPE, and MAE. The following are the most significant research contributions of this study:

1. This study introduces the EMAGRU model's architecture for accurate and reliable stock price prediction.
2. We introduce AntiReLU and combine it with ReLU as the activation function in the EMAGRU layers.
3. We evaluate how well the proposed model performs compared to other models.

These results can be interesting for other researchers and institutions because the proposed model provides a new approach to predicting stock prices accurately and reliably, which can help investors and businesses maximize profits and reduce risks. The EMAGRU model's architecture and the combination of AntiReLU and ReLU activation functions can also be used in other deep learning models for time series analysis.

2. METHOD

2.1. Data Preprocessing

In this research, stock trading data of PT is used. Telkom Indonesia (Persero) Tbk with the stock symbol TLKM.JK for ten years from October 30, 2012, to October 30, 2022, as shown in Figure 1. The historical data came from Yahoo Finance. The records have six attributes: Open, High, Low, Close, Adj Close, and Volume.

At this stage, the authors remove rows with a volume of 0 and those with missing values or NANs. The formed dataset has 2410 records. In this study, only close price data are used. Then, the authors create two representations of averaged data over a 10 and 20-day period by calculating EMA10 and EMA20 based on Equation (1).

$$EMA = Price(t) \times k + EMA(y) \times (1 - k) \tag{1}$$

where t = today, y = yesterday, N = number of days, and $k = 2 \div (N + 1)$. Two EMAs with different N generate buy and sell signals based on moving average crossovers and divergences.

Figure 2 visualizes EMA10 and EMA20, representing the trend of daily stock close price movements for ten years. The blue color represents EMA10, and the red color represents EMA20. The trend is said to be up when the blue color is above the red color. Conversely, the trend declines when the red color is above the blue color.

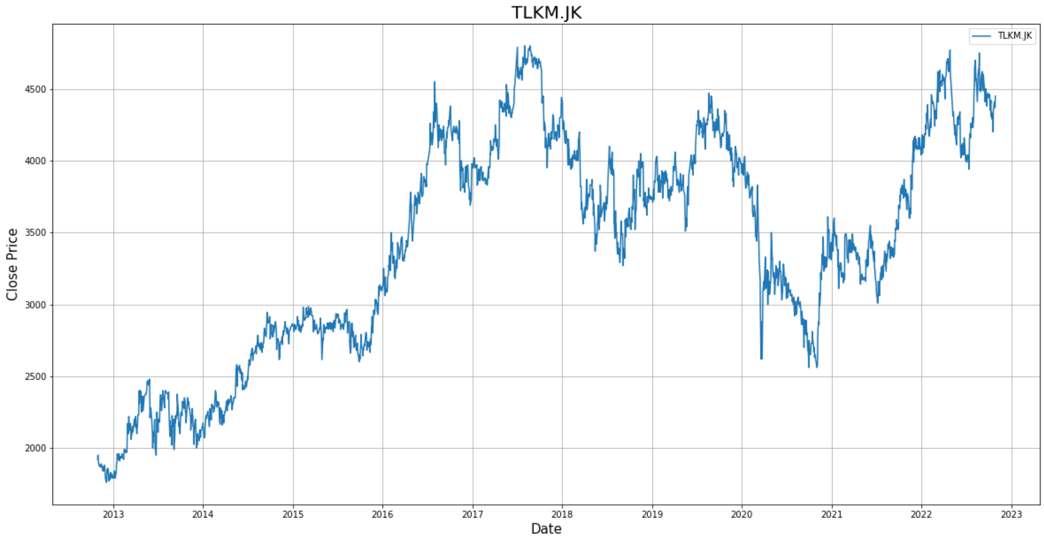

Fig. 1. Stock chart TLKM.JK from October 30, 2012, until October 30, 2022. Data from Yahoo Finance

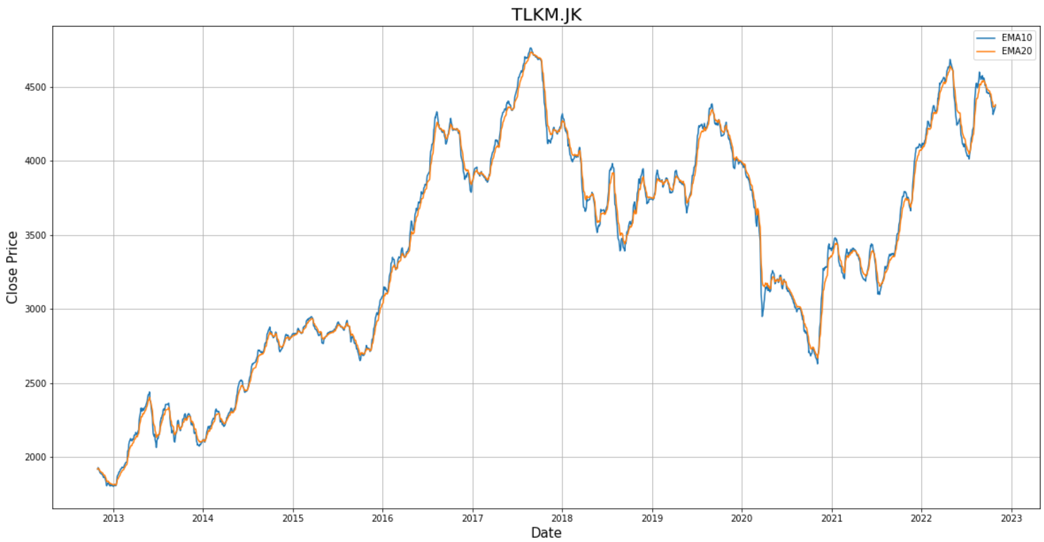

Fig. 2. Stock price representation based on EMA10 and EMA20 calculation

For each EMA, the minimum value is transformed into 0 and the maximum value into 1. Equation (2) converts minimum and maximum values to 0 and 1.

$$x_{norm} = \frac{x - x_{min}}{x_{max} - x_{min}} \tag{2}$$

2.2. Data Splitting

There are two halves of information for each EMA, for 2410. In the first half, the authors introduce the 2110 data for training (65%) and validation data (35%). The second half is a sample size of 300 for evaluation purposes.

2.3. Activation Function

In this research, the ReLU and AntiReLU activation functions are used. When ReLU calculates that every input less than zero results in an output of 0, and every other input x produces an output of x as well, AntiReLU calculates the opposite. For every input x that is greater than 0, AntiReLU gives an output of 0. Conversely, when the input is negative, the output of AntiReLU is also negative. Figure 3 shows the blue line as the representation of ReLU and the green line as AntiReLU.

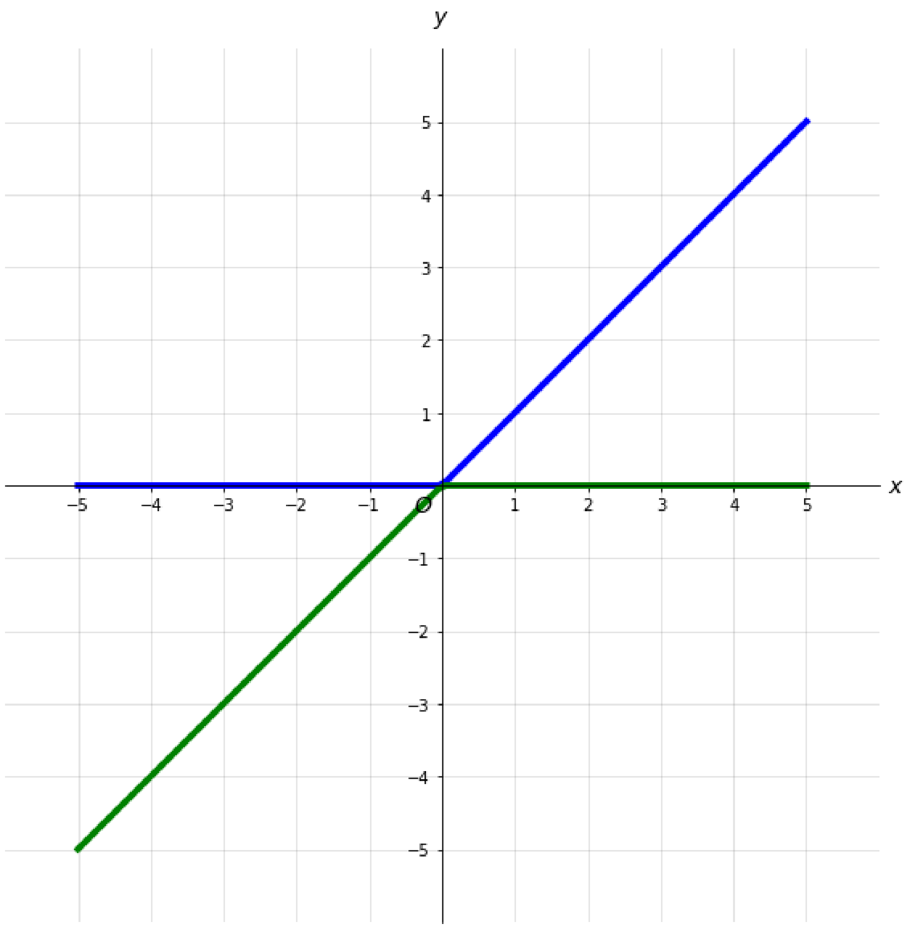

Figure 3. ReLU (blue line) and AntiReLU (green line).

2.4. Gated Recurrent Unit and EMAGRU

A Gated Recurrent Unit (GRU) is a more recent Recurrent Neural Network analogous to LSTM. A Gated Recurrent Unit is responsible for eliminating the cell state and exchanging information utilizing the hidden state. Unlike the LSTM, GRU has a reset and an update gate. The update gate functions similarly to the forget and input gates in LSTM, choosing what information to keep and what to throw out or add. Forgetting previous information is accomplished through the reset gate.

The following Equation (3) describes how GRU computes the reset gate R_t and forget gate Z_t.

$$\begin{aligned} R_t &= \sigma(X_t\, W_{xr} + H_{t-1}\, W_{hr} + b_r) \\ Z_t &= \sigma(X_t\, W_{xz} + H_{t-1}\, W_{hz} + b_z) \end{aligned} \tag{3}$$

where W is the weight parameter, and b is the bias parameter. Candidate hidden state $\widetilde{H}_t$ at time step t is computed in Equation(4).

$$\widetilde{H}_t = tanh(X_t\, W_{xh} + (R_t \times H_{t-1})\, W_{hh} + b_h) \tag{4}$$

The impact of updating gate Z_t must be accounted for at last. This establishes how closely the new hidden state H_t resembles the previous state H_{t-1}, in comparison to the candidate state $\widetilde{H}_t$. An elementwise convex combination of H_{t-1} and $\widetilde{H}_t$ can be used in place of the update gate Z_t to achieve this goal, as shown in Equation (5).

$$H_t = (1 - Z_t) \times \widetilde{H}_t + Z_t \times H_{t-1} \tag{5}$$

EMAGRU is a GRU model with two inputs, namely EMA10 and EMA20. We parallelize two stacked GRUs, A and B, and serialize 2 GRU layers in each stacked GRU, as shown in Figure 4.

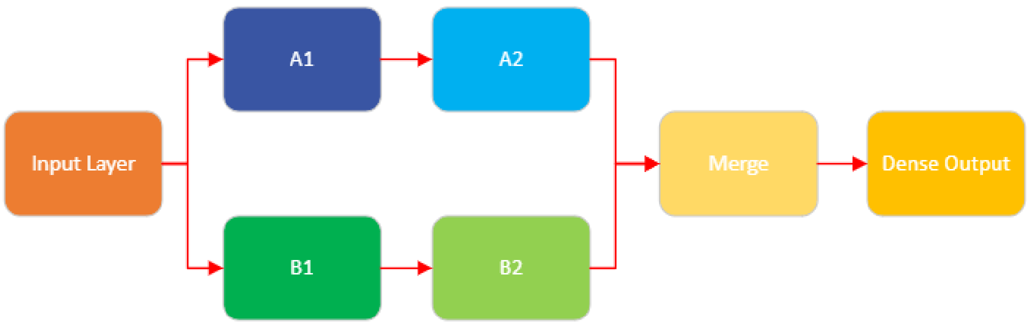

Fig. 4. Proposed EMAGRU Workflow

We place ReLU or AntiReLU activation functions on each layer. With this combination, we propose 6 EMAGRU models, as shown in Table 1.

Tab. 1. Proposed EMAGRU Architecture

Model	A1	A2	B1	B2
M1	ReLU	ReLU	ReLU	ReLU
M2	AntiReLU	AntiReLU	AntiReLU	AntiReLU
M3	ReLU	AntiReLU	ReLU	AntiReLU
M4	AntiReLU	ReLU	AntiReLU	ReLU
M5	ReLU	ReLU	AntiReLU	AntiReLU
M6	ReLU	AntiReLU	AntiReLU	ReLU

2.5. Parameter Settings

The proposed model predicts the next day's stock prices based on the previous 50 days of input data. After the input is entered into the GRU hidden layer, the input data is spread over the 300 neurons. This model disseminates its forecasts for the following day using EMA10 and EMA20 through a dense layer.

The proposed model uses the Adam optimizer, loss measurement of mean square error (*MSE*), and accuracy metrics. During training, the model uses batch 32 and epoch 100. Table 2 presents the parameter settings, and Table 3 shows the detailed parameters used in the EMAGRU model.

Tab. 2. Parameter Settings.

Parameter	Value
n_step	50
layer_neurons	300
n_outs	2
optimizer	Adam
loss	mse
metrics	accuracy
batch	32
epoch	100

Tab. 3. Detailed parameters of EMAGRU

Layer (Type)	Output Shape	# Parameters	Connected to
Input (InputLayer)	[(None, 50, 2)]	0	0
A1 (GRU)	(None, 50, 300)	273600	Input
B1 (GRU)	(None, 50, 300)	273600	Input
A2 (GRU)	(None, 300)	541800	A1
B2 (GRU)	(None, 300)	541800	B1
Merge (Concatenate)	(None, 600)	0	A2, B2
Output (Dense)	(None, 2)	1202	Merge
Total Parameters:		1.632.002	
Trainable Parameters:		1.632.002	

2.6. Model Evaluation

In our study, the predictive performance of the model is evaluated using Root Mean Square Error (RMSE), Mean Absolute Percentage Error (MAPE), Mean Absolute Error (MAE), and coefficient of determination R^2 calculated according to Equations (6) - (9).

$$RMSE = \sqrt{\frac{1}{n}\sum_{t=1}^{n}(y_t - \hat{y}_t)^2} \tag{6}$$

$$MAE = \frac{1}{n}\sum_{t=1}^{n}|y_t - \hat{y}_t| \tag{7}$$

$$MAPE = \frac{1}{n}\sum_{t=1}^{n}\frac{|y_t - \hat{y}_t|}{|y_t|} \tag{8}$$

$$R^2 = 1 - \frac{SS_{RES}}{SS_{TOT}} = 1 - \frac{\sum_{t=1}^{n}(y_t - \hat{y}_t)^2}{\sum_{t=1}^{n}(y_t - \bar{y}_t)^2} \tag{9}$$

3. RESULTS AND DISCUSSION

3.1. Training and Validation Accuracy

Based on Table 4 presented in the research paper, it's apparent that six different models, denoted as M1 to M6 are compared and used for stock prediction. The comparison is based on the performance of these models in terms of their Training Accuracy and Validation Accuracy. Training Accuracy refers to the model's performance on the same dataset it was trained on, while Validation Accuracy represents how well the model generalizes to unseen data from a validation set. Ideally, we want both accuracies to be high, but the Validation Accuracy is often considered more critical because it indicates the model's ability to handle new, unseen data.

Tab. 4. Training and Validation Accuracy.

Model	Training Accuracy	Validation Accuracy
M1	0.9413	0.9760
M2	0.9486	0.9798
M3	0.9433	**0.9823**
M4	0.9466	**0.9823**
M5	**0.9519**	0.9811
M6	0.9466	0.9786
Average	0.9464	0.9800

Model M1 has a Training Accuracy of 0.9413 and a Validation Accuracy of 0.9760. These numbers imply that M1 can accurately predict the training data about 94.13% of the time and can accurately predict the validation data around 97.60% of the time. Model M2 performs slightly better than M1 with a Training Accuracy of 0.9486 and a Validation Accuracy of 0.9798. This suggests the M2 model can generalize better to unseen data than the M1. Model M3 and M4 have the same Validation Accuracy of 0.9823, the highest among the models, showing that they are the best at generalizing to new data. Their Training Accuracy is also relatively high, at 0.9433 for M3 and 0.9466 for M4, indicating a good fit on the training data. Model M5, while having the highest Training Accuracy of 0.9519, slightly underperforms in Validation Accuracy (0.9811) compared to M3 and M4. This might suggest some overfitting, as it performs best on the training data but falls slightly short on unseen data. Finally, Model M6 shows a good balance between Training Accuracy (0.9466) and Validation Accuracy (0.9786) but doesn't excel either.

Looking at the average performance, we see a Training Accuracy of 0.9464 and a Validation Accuracy of 0.9800. These average figures demonstrate that overall, the models perform well on training and unseen data, but there is some variation in performance among the different models. Regarding overall reliability and robustness, M3 and M4 perform best, offering the highest validation accuracies, indicating superior generalization capabilities. However, the specific choice of model could depend on other factors such as computational efficiency, complexity, interpretability, and more.

The accuracy of the learning set was lower than the validating set, which may suggest that other learning parameters should be used. There are several possible reasons why the accuracy of the learning set was lower, including overfitting, learning rate, and number of epochs. Overfitting occurs when the model learns the training data too well and cannot generalize to new data, leading to high accuracy on the validating set but lower accuracy on the learning set. The learning rate determines how quickly the model learns from the data, and if it is too high or too low, it can affect the model's ability to understand the underlying patterns in the data. The number of epochs determines how often the model sees the entire training data, and if it is too low or too high, it can affect the model's ability to learn the underlying patterns in the data.

3.2. Prediction of the next 20 days

After successfully making predictions for the next day, the authors extrapolate predictions for the following 20 trading days. Based on the EMA10 and EMA20 predictions, Figure 5 shows that in the next 20 days, the market will tend to decline.

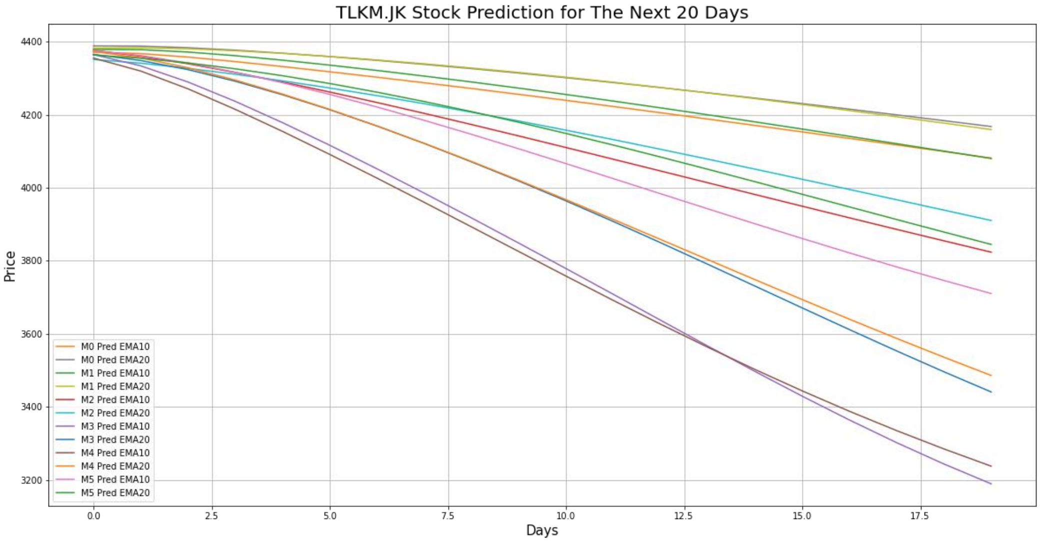

Fig. 5. EMA10 and EMA20 predictions for the next 20 days

3.3. Performance Metrics

Table 5 shows the EMAGRU Performance of EMA10. Model M2 and M6 offer the best performance with the lowest RMSE, MAPE, MAE, and highest R^2, indicating they predict the EMA10 dataset more accurately than the other models. In contrast, Model M5 performs the least well across all metrics, implying it may not be the best choice for predicting the EMA10 dataset.

Tab. 5. EMAGRU Performance of EMA10

Model	RMSE	MAPE	MAE	R^2
M1	0.0054	0.0072	0.0056	0.9973
M2	**0.0045**	0.0044	**0.0034**	0.9988
M3	0.0048	0.0051	0.0040	0.9984
M4	0.0077	0.0082	0.0065	0.9965
M5	0.0094	0.0092	0.0072	0.9958
M6	0.0049	**0.0043**	**0.0034**	**0.9989**
Average	0.0060	0.0064	0.0050	0.9976

Table 6 shows the EMAGRU Performance of EMA20. Model M3 stands out with the lowest RMSE, MAPE, and MAE, and the highest R^2, indicating it performs best on the EMA20. Model M1 performs the worst on the EMA20 across all metrics, suggesting it may be less suitable for this EMA.

Tab. 6. EMAGRU Performance of EMA20

Model	RMSE	MAPE	MAE	R^2
M1	0.0089	0.0111	0.0085	0.9953
M2	0.0050	0.0058	0.0045	0.9985
M3	**0.0025**	**0.0025**	**0.0020**	**0.9996**
M4	0.0034	0.0037	0.0029	0.9993
M5	0.0068	0.0083	0.0064	0.9972
M6	0.0031	0.0034	0.0026	0.9994
Average	0.0050	0.0058	0.0045	0.9982

Figure 6 shows the result of a comparison between the actual values and the EMA10 and EMA20 predictions for the last 300 days.

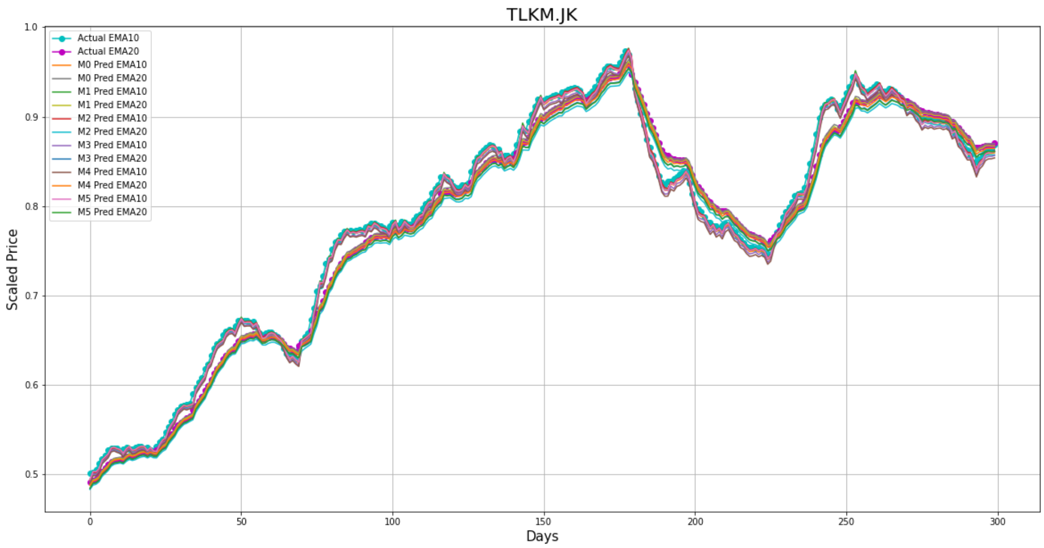

Fig. 6. Performance of EMA10 and EMA20 Predictions

3.4. Comparison to the previous studies

Among the results of several previous studies, EMAGRU can be a reliable alternative for predicting stock prices. Table 7 below compares models, and generally, the proposed EMAGRU produces better predictions indicated by the RMSE, MAPE, MAE, and R^2 metrics.

Tab. 7. Model Comparison

Model	RMSE	MAPE	MAE	R^2
LSTM (Lv et al., 2021)	0.0105		0.0065	
Doc-W-LSTM (Ji et al., 2021)	0.1100		0.0190	0.9570
CNN-LSTM (Lu et al., 2020)	0.3969		0.2756	0.9646
WLSTM+Att (Qiu et al., 2020)	0.1971		0.1569	0.9621
LSTM-BO-XGB (Liwei et al., 2021)	0.0610		0.1560	
LSTM (Nabipour et al., 2020)	0.0065	0.4300	0.4460	
LSTM BGA8X13 (Wang et al., 2021)	1.1345	0.0020		
LSTM (DVA) (Ta et al., 2020)			0.0597	
LASSO-LSTM (Gao et al., 2021)	0.2709		0.1880	
PCA-LSTM (Gao et al., 2021)	0.6368		0.4289	
GAN (Diqi et al., 2022)			0.0207	0.8112
GAN (Lin et al., 2021)	0.0533			
GRU-Only (Shahi et al., 2020)	0.4731		0.4281	0.8790
GRU-News (Shahi et al., 2020)	0.2915		0.2447	0.9670
LASSO-GRU(Gao et al., 2021)	0.2815		0.1986	
PCA-GRU (Gao et al., 2021)	0.8121		0.6132	
GRU (DVA) (Ta et al., 2020)			0.0473	
GRU (S&P500) (Zhang & Fang, 2021)	0.0843		0.0710	
GRU (Manjunath et al., 2021)	0.1200		0.1060	
Proposed EMAGRU	**0.0050**	**0.0058**	**0.0045**	**0.9982**

4. CONCLUSION

Investors and company owners want reliable stock price predictions to maximize earnings and minimize losses. In this research, a GRU is used as the primary model. EMAGRU predicts the following day's stock price using the EMA10 and EMA20 inputs from the preceding 50 days. Predictions for the next 20 days can be used to build the future trend of whether the stock price will go up, down, or flat. Table 7 shows that EMAGRU has the highest performance among numerous other models predicting stock prices. When looking at EMA10's performance, the average values for RMSE, MAPE, MAE, and R^2 are 0.0060, 0.0064, 0.0050, and 0.9976, respectively. The values for the RMSE, MAPE, MAE and R^2 for the EMA20 are 0.0050, 0.0058, 0.0045, and 0.9982, respectively. Combining ReLU and AntiReLU activation functions in the EMAGRU model may have helped the model learn the data's underlying patterns more effectively. Additionally, using batch normalization and dropout techniques prevented overfitting and contributed to the better performance of the EMAGRU model. Overall, the proposed EMAGRU model is a novel and practical approach to stock price prediction that outperforms other models regarding accuracy and reliability. The authors suggest using the EMAGRU of other equities or stock indices or exploring the impact of varying hyperparameters as potential future work.

Acknowledgments

This paper is conducted in the Department of Information Systems and the Department of Informatics, Universitas Respati Yogyakarta, Yogyakarta, Indonesia.

Conflicts of Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper. On behalf of all authors, the corresponding author states that there is no conflict of interest.

REFERENCES

Chen, W., Jiang, M., Zhang, W. G., & Chen, Z. (2021). A novel graph convolutional feature based convolutional neural network for stock trend prediction. *Information Sciences*, *556*, 67-94. https://doi.org/10.1016/j.ins.2020.12.068

Chen, W., Zhang, H., Mehlawat, M. K., & Jia, L. (2021). Mean–variance portfolio optimization using machine learning-based stock price prediction. *Applied Soft Computing, 100,* 106943. https://doi.org/10.1016/j.asoc.2020.106943

Chiniforoush, N., & Latif Shabgahi, G. (2021). A novel method for forecasting surface wind speed using wind-direction based on hierarchical markov model. *International Journal of Engineering, 34*(2), 414-426. https://doi.org/10.5829/ije.2021.34.02b.13

Chun, J., Ahn, J., Kim, Y., & Lee, S. (2021). Using deep learning to develop a stock price prediction model based on individual investor emotions. *Journal of Behavioral Finance*, *22*(4), 480-489. https://doi.org/10.1080/15427560.2020.1821686

Diqi, M. (2022). StockTM: Accurate stock price prediction model using LSTM. *International Journal of Informatics and Computation*, *4*(1), 1-10. https://doi.org/10.35842/ijicom.v4i1.50

Diqi, M., Hiswati, M. E., & Nur, A. S. (2022). StockGAN: Robust stock price prediction using GAN algorithm. *International Journal of Information Technology*, *14*(5), 2309–2315. https://doi.org/10.1007/s41870-022-00929-6

Diqi, M., Mulyani, S. H., & Pradila, R. (2023). DeepCov: Effective prediction model of COVID-19 using CNN algorithm. *SN Computer Science, 4,* 396. https://doi.org/10.1007/s42979-023-01834-w

Gao, Y., Wang, R., & Zhou, E. (2021). Stock prediction based on optimized LSTM and GRU models. *Scientific Programming, 2021,* 4055281. https://doi.org/10.1155/2021/4055281

Jabeen, A., Afzal, S., Maqsood, M., Mehmood, I., Yasmin, S., Niaz, M. T., & Nam, Y. (2021). An LSTM based forecasting for major stock sectors using COVID sentiment. *Computers, Materials and Continua*, *67*(1), 1191-1206. https://doi.org/10.32604/cmc.2021.014598

Ji, X., Wang, J., & Yan, Z. (2021). A stock price prediction method based on deep learning technology. *International Journal of Crowd Science*, *5*(1), 55-72. https://doi.org/10.1108/IJCS-05-2020-0012

Jin, Z., Yang, Y., & Liu, Y. (2020). Stock closing price prediction based on sentiment analysis and LSTM. *Neural Computing and Applications*, *32*(13), 9713-9729. https://doi.org/10.1007/s00521-019-04504-2

Khan, W., Ghazanfar, M. A., Azam, M. A., Karami, A., Alyoubi, K. H., & Alfakeeh, A. S. (2022). Stock market prediction using machine learning classifiers and social media, news. *Journal of Ambient Intelligence and Humanized Computing*, *13*(7), 3433-3456. https://doi.org/10.1007/s12652-020-01839-w

Ko, C.- R., & Chang, H.- T. (2021). LSTM-based sentiment analysis for stock price forecast. *PeerJ Computer Science*, *7*, e408. https://doi.org/10.7717/peerj-cs.408

Li, X., Wu, P., & Wang, W. (2020). Incorporating stock prices and news sentiments for stock market prediction: A case of Hong Kong. *Information Processing and Management*, *57*(5), 102212. https://doi.org/10.1016/j.ipm.2020.102212

Lin, H.- C., Chen, C., Huang, G.- F., & Jafari, A. (2021). Stock price prediction using generative adversarial networks. *Journal of Computer Science*, *17*(3), 188-196. https://doi.org/10.3844/JCSSP.2021.188.196

Liwei, T., Li, F., Yu, S., & Yuankai, G. (2021). Forecast of LSTM-XGBoost in stock price based on bayesian optimization. *Intelligent Automation and Soft Computing*, *29*(3), 855-868. https://doi.org/10.32604/iasc.2021.016805

Lu, W., Li, J., Li, Y., Sun, A., & Wang, J. (2020). A CNN-LSTM-based model to forecast stock prices. *Complexity, 2020,* 6622927. https://doi.org/10.1155/2020/6622927

Lv, J., Wang, C., Gao, W., & Zhao, Q. (2021). An Economic Forecasting Method Based on the LightGBM-Optimized LSTM and Time-Series Model. *Computational Intelligence and Neuroscience*, *2021*, 8128879. https://doi.org/10.1155/2021/8128879

Manjunath, C., Marimuthu, B., & Ghosh, B. (2021). Deep learning for stock market index price movement forecasting using improved technical analysis. *International Journal of Intelligent Engineering and Systems*, *14*(5), 129-141. https://doi.org/10.22266/ijies2021.1031.13

Nabipour, M., Nayyeri, P., Jabani, H., Mosavi, A., Salwana, E., & Shahab, S. (2020). Deep learning for stock market prediction. *Entropy*, *22*(8), 840. https://doi.org/10.3390/E22080840

Nti, I. K., Adekoya, A. F., & Weyori, B. A. (2020). A systematic review of fundamental and technical analysis of stock market predictions. *Artificial Intelligence Review*, *53*(4), 3007-3057. https://doi.org/10.1007/s10462-019-09754-z

Qiu, J., Wang, B., & Zhou, C. (2020). Forecasting stock prices with long-short term memory neural network based on attention mechanism. *PLoS ONE*, *15*(1), e0227222. https://doi.org/10.1371/journal.pone.0227222

Radojičić, D., & Kredatus, S. (2020). The impact of stock market price fourier transform analysis on the gated recurrent unit classifier model. *Expert Systems with Applications, 159,* 113565. https://doi.org/10.1016/j.eswa.2020.113565

Saud, A. S., & Shakya, S. (2021). 3-Way gated recurrent unit network architecture for stock price prediction. *Indian Journal of Computer Science and Engineering*, *12*(2), 421-427. https://doi.org/10.21817/indjcse/2021/v12i2/211202011

Savadi Hosseini, M., & Ghaderi, F. (2020). A Hybrid deep learning architecture using 3D CNNs and GRUs for human action recognition. *International Journal of Engineering*, *33*(5), 959-965. https://doi.org/10.5829/ije.2020.33.05b.29

Shahi, T. B., Shrestha, A., Neupane, A., & Guo, W. (2020). Stock price forecasting with deep learning: a comparative study. *Mathematics*, *8*(9), 1441. https://doi.org/10.3390/math8091441

Ta, V.- D., Liu, C.- M., & Tadesse, D. A. (2020). Portfolio optimization-based stock prediction using long-short term memory network in quantitative trading. *Applied Sciences*, *10*(2), 437. https://doi.org/10.3390/app10020437

Thormann, M.- L., Farchmin, J., Weisser, C., Kruse, R.- M., Safken, B., & Silbersdorff, A. (2021). Stock price predictions with LSTM neural networks and twitter sentiment. *Statistics, Optimization and Information Computing*, *9*(2), 268-287. https://doi.org/10.19139/soic-2310-5070-1202

Wang, C.- C., Chien, C.- H., & Trappey, A. J. C. (2021). On the application of ARIMA and LSTM to predict order demand based on short lead time and on-time delivery requirements. *Processes*, *9*(7), 1157. https://doi.org/10.3390/pr9071157

Zhang, S., & Fang, W. (2021). Multifractal behaviors of stock indices and their ability to improve forecasting in a volatility clustering period. *Entropy*, *23*(8), 1018. https://doi.org/10.3390/e23081018

Zhao, J., Zeng, D., Liang, S., Kang, H., & Liu, Q. (2021). Prediction model for stock price trend based on recurrent neural network. *Journal of Ambient Intelligence and Humanized Computing*, *12*, 745-753. https://doi.org/10.1007/s12652-020-02057-0