International Journal of Engineering Trends and Technology
ISSN: 2231–5381 / https://doi.org/10.14445/22315381/IJETT-V71I6P215
Volume 71 Issue 6, 126-133, June 2023

Original Article

Time Series Forecasting Based on Deep Learning CNN-LSTM-GRU Model on Stock Prices

Ghani Rizky Naufal[1], Antoni Wibowo[2]

[1,2]*Computer Science Department, School of Computer Science, Bina Nusantara University, Jakarta, Indonesia.*

[1]*Corresponding Author : ghani.naufal@binus.ac.id*

Received: 10 April 2023 Revised: 10 June 2023 Accepted: 15 June 2023 Published: 25 June 2023

Abstract *- An equity, commonly referred to as a stock, is a financial asset representing partial ownership in the company that issued it. Stocks are favored among investors due to their potential for quick financial gains. It is important to note that stock prices are not arbitrary but instead follow specific patterns and can be analyzed and predicted using discrete time series models. This allows for studying and forecasting stock prices, enabling investors to make informed decisions. Much research has been done to forecast stock prices. The Long Short-Term Memory (LSTM) model, known for its popularity in deep learning, lacks the necessary strength to serve as the main approach for time series forecasting. To address the drawbacks, many various elements employing many methodologies and procedures are addressed; one option is to integrate several deep learning models to create a hybrid deep learning model. Researchers have utilized hybrid deep learning models LSTM-GRU and CNN-LSTM to forecast and analyze time series data, and both models outperform LSTM. The purpose of this research is to create a hybrid deep learning model that combines deep neural network-based models LSTM and Gated Recurrent Units (GRU) with 1D Convolutional Neural Networks (1D-CNN) that focuses on three stock prices, TSLA, GOOG, and TWTR. The findings reveal that the proposed model outperforms the LSTM network and other hybrid models in terms of prediction errors. This shows that the proposed model is effective in forecasting stock prices.*

Keywords *- Deep learning, Time series, Forecasting, Stock prices, CNN, LSTM, GRU.*

1. Introduction

Stocks or equity are financial instruments that represent ownership in a company. When an individual buys shares of a stock, they are entitled to a proportionate amount of the company's assets and earnings. These shares are commonly traded on stock exchanges and are an essential component of many investors' portfolios.

Stock price forecasts are usually popular due to the quick cash gains and affiliated complexities. Stock prices are not created randomly but may be viewed as discrete time series models, and patterns can be carefully examined, allowing them to be forecasted. One reason for stock forecasting is financial advantage. In a dynamic stock market, a system that can determine which stocks are performing well and which are not makes it simpler for investors, markets, or financial experts to make judgments.

A time series model analysis differs from other data analysis methods in that it may show how variables change over time. In other words, time is a crucial variable since it demonstrates how data changes through time and the results. It provides an additional source of information and a preset sequence of data dependencies. Time series analysis usually needs a large number of data points in order to maintain consistency and dependability. Time series models are more accurate than explanatory or mixed models in forecasting [1]. Because of the volatility of stock prices, it is critical to find suitable approaches and models for estimating their values.

Many studies have been conducted for time series forecasting. The most used classical technique for time series forecasting is the autoregressive integrated moving average (ARIMA). ARIMA outperformed other traditional time series forecasting approaches such as univariate Autoregressive (AR), univariate Moving Average (MA), and Simple Exponential Smoothing (SES).

Nevertheless, the increasing popularity of Deep Learning techniques in the field of financial and economic time series analysis and prediction is growing. Specifically, the Long Short-Term Memory (LSTM) model has garnered considerable attention and, in certain instances, has outperformed the ARIMA model [2].

Deep learning's popularity can be attributed to three factors, vastly improved processing capabilities, dramatically increased data size used for training, and recent advances in machine learning that allow deep learning to use complex, nonlinear compositional functions, learn distributed and hierarchical feature representations, and efficiently use of labelled and unlabelled data [3].

However, in a study by Yamak et al. [5], LSTM is not powerful enough to be employed as a primary forecasting approach for time series forecasting. In this study, ARIMA beats LSTM when the data used is few, but GRU outperforms LSTM. Many studies have been conducted to address the flaws in LSTM, and one approach is to combine different deep learning models to cover the weaknesses in deep learning models, making it a hybrid deep learning model.

In a previous study conducted by Patel et al. [6], a hybrid model combining LSTM and GRU was developed to address the shortcomings of each model. The aim was to rectify the underperformance of LSTM and the tendency of GRU to over-predict results [7]. The proposed hybrid model demonstrated superior performance compared to LSTM alone. Similarly, another study introduced a hybrid deep learning model CNN-LSTM to overcome the limitations of LSTM when dealing with limited data for analysis [9,10,12,13]. These hybrid approaches exhibited significant performance improvements and outperformed the standalone LSTM model.

Despite these advancements, there exists a research gap in the development of a hybrid deep learning model that surpasses the performance of LSTM alone. While previous studies have explored hybrid models such as LSTM-GRU and CNN-LSTM, incorporating CNN with the LSTM-GRU model is unexplored. This study aims to propose a hybrid deep learning model that combines LSTM, GRU, and 1D Convolutional Neural Networks (1D-CNN) for concurrent feature extraction and input categorization [15]. The model's objective is to analyse and predict stock market data based on historical information. By harnessing the strengths of these deep neural network models, the research aims to improve forecasting accuracy and support decision-making in the context of stock market investments through historical data analysis.

2. Literature Survey

A significant amount of study has been carried out for time series forecasting. Many different factors utilizing multiple approaches and processes in Machine Learning and Deep Learning have been considered. Yamak et al. [3] conducted a comprehensive analysis that considered a wide range of factors, incorporating multiple approaches and processes within the fields of Machine Learning and Deep Learning. Specifically, their study involved a comparison of three distinct modes ARIMA, LSTM, and GRU. According to the findings, the results indicated that the ARIMA model exhibited superior performance compared to the deep learning-based regression methods LSTM and GRU. However, it is worth noting that GRU outperformed LSTM, achieving a lower Mean Absolute Percentage Error (MAPE) of 3.97 percent and a Root Mean Square Error (RMSE) of 381.34. Several factors could contribute to these results, including the limited amount of data utilized in the study, which may have favoured the performance of ARIMA over the other models.

In a study conducted by Patel et al. [6], a prediction scheme based on a hybrid LSTM-GRU model is proposed. The motivation behind this hybrid model is to address the limitations of each individual model, where LSTM may underperform and GRU may over-predict results. The input for both models is similar, and their networks are combined and passed through a dense layer to obtain the final output. The study results demonstrate that the proposed hybrid scheme achieves highly accurate price predictions and outperforms the standalone LSTM model in terms of performance.

Table 1. Summary of related works

Authors	Algorithms / Models	Result
Yamak et al., 2019 [5]	ARIMA, LSTM, GRU	ARIMA outperform LSTM and GRU, while GRU outperforms LSTM in performance.
Patel et al., 2020 [6]	LSTM-GRU, LSTM	LSTM-GRU outperform LSTM in performance
Livieris et al., 2021 [9]	CNN-LSTM	Efficiently utilizing mixed time-series data avoids overfitting and reduces computing costs when compared to typical fully connected deep neural networks.
Alonso-Monsalve et al., 2020 [10]	CNN-LSTM, MLP, RBFN, CNN	Convolutional LSTM clearly outperforms the rest of all other models.
Lu et al., 2020 [12]	CNN-LSTM	CNN-LSTM yields reliable stock price forecasting with scores of 27.564 MAE, 39.688 RMSE, and 0.9646 R2.
Vidal& Kristjanpoller [13]	CNN-LSTM, GARCH, LSTM	CNN-LSTM model's MSE is decreased by 37% when compared to the regular GARCH model and by 18% when compared to the LSTM model.

The CNN-LSTM approach also has gained significant attention in the research community, with numerous researchers exploring its application in time series forecasting. Specifically, studies by Alonso-Monsalve et al., Livieris et al., and Pintelas et al. [9,10,12,13] have extensively employed CNN-LSTM for this purpose. These studies recognized the potential of CNN in removing noise from raw input data and extracting meaningful features, thereby facilitating the handling of input data for LSTM.

By combining the capabilities of CNN and LSTM, these hybrid techniques aimed to capture both long and short-sequence pattern dependencies to make accurate predictions, particularly in the context of forecasting time series values. The results obtained from these studies consistently demonstrated the superiority of the hybrid CNN-LSTM approach over the conventional LSTM model. The incorporation of CNN as a pre-processing step significantly improved the quality of the input data, enabling LSTM to effectively capture the intricate temporal relationships inherent in the time series data. The combination of CNN and LSTM yielded enhanced forecasting accuracy and outperformed the standalone LSTM model in terms of prediction performance.

The existing literature has not extensively explored using CNN in combination with LSTM-GRU models. By incorporating CNN, a specialized form of convolutional neural network designed for processing one-dimensional inputs like time series data, the model aims to filter noise and extract meaningful features effectively. Additionally, the integration of LSTM and GRU enhances the model's performance by addressing both short-term and long-term dependencies. LSTM excels at capturing long-term dependencies, while GRU offers faster training and convergence, albeit with a potential trade-off in long-term dependency capture. By emphasizing these distinctive features and comparing them with previous research, this study showcases its novelty and potential to improve the accuracy and efficiency of stock price forecasting.

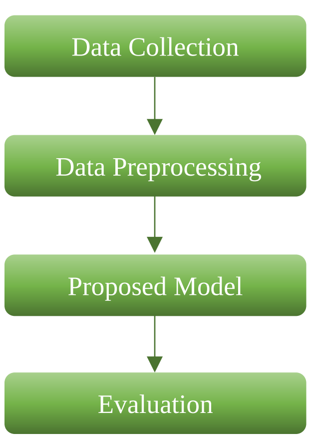

Fig. 1 Research Process

Table 2. Example of dataset features

Date	Open	High	Low	Close	Adj Close
2022-10-03	254.5	255.16	241.009	242.399	242.399
2022-10-04	250.52	257.5	242.009	249.440	249.440
2022-10-05	245.009	246.669	233.270	240.809	240.809
2022-10-06	239.44	244.58	235.350	238.130	238.130

3. Proposed Methods

This section offers a comprehensive discussion of the research process outlined in Figure 1, encompassing various steps, including data collection, data pre-processing, development of the proposed model, and subsequent evaluation. The evaluation of the proposed model will include a comparative analysis with two other models, namely LSTM and LSTM-GRU, to ascertain whether the proposed model outperforms the comparison models.

3.1. Data Collection

In this study, the historical stock price data from three prominent companies, Tesla (TSLA), Google (GOOG), and Twitter (TWTR), was collected and utilized for analysis. The data was sourced from the reputable finance website https://finance.yahoo.com, which provides reliable and up-to-date information on financial markets and stock prices.

The data collection spanned over a period of five years, starting from October 9, 2017, and ending on October 7, 2022, yielding a total of 1258 data points. The collected data consisted of seven features, as shown in Table 2, namely date, open, high, low, close, and adjusted close, providing a comprehensive and detailed picture of the selected companies' stock prices and trading activities over the specified time frame.

3.2. Data Pre-Processing

The stock price data will undergo pre-processing before being used for analysis. The pre-processing step plays a critical role as it is essential in ensuring that the data is cleaned and prepared for analysis. Firstly, any empty or null data points were removed from the data using the panda's data frame. Once the missing data had been removed, the data were narrowed down to only include the date and close columns. These columns are considered the most relevant features for the analysis, as they provide important information about the dates when the stock prices were recorded and the corresponding closing prices.

The dataset is subsequently split into three portions 80% for training, 20% for testing, and 10% of the training data reserved for validation. The validation set plays a crucial role in preventing overfitting and accurately assessing the model's performance. The Min-Max normalization function from the sklearn library is employed to ensure data consistency. Moreover, the data is transformed into input and output pairs using two distinct window sizes, 1 and 30. This technique allows the model to capture both short-term and long-term trends in stock prices [3].

$$x = [p_{n-w+1}, p_{n-w+2}, \ldots, p_{n-1}, p_n] \quad (1)$$

$$y = [p_{n+1}] \quad (2)$$

In the given equation, the symbols represent the following, "x" represents the input data, "y" represents the output data, and "Pn" represents the prices at a specific time stamp "n". Separating data into input and output pairs is critical as it allows predicting future values based on historical trends and patterns. The input data consists of prices leading up to the time stamp "n", while the output data consists of prices for a subsequent time after the time stamp "n+1". This division enables the model to learn from past prices and generate forecasts for future prices.

3.3. Proposed Model

This study introduces a hybrid deep learning model, as shown in Figure 2, which combines 1D Convolutional Neural Networks (1D-CNN), Long Short-Term Memory (LSTM), and Gated Recurrent Unit (GRU). The research evaluates time series data and generates forecasts based on previous data analysis, specifically targeting TSLA, GOOG, and TWTR data. Along with assessing the performance of proposed hybrid model, the study aims to compare its effectiveness against other learning models, such as LSTM and LSTM-GRU. This comparative analysis will help determine the strengths and weaknesses of each model and provide valuable insights into its suitability for different types of data and applications.

All the architecture will have the same input gained from the data pre-processing step, with a window size of 1 and 30 that are calculated based on equations 1 and 2. The window size refers to the number of past time steps utilized to forecast the value at the current time step.

A window size of 1 implies that only the previous time step is considered for prediction, whereas a window size of 30 incorporates the previous 30-time steps. A window size of 1 focuses solely on immediate past information, while a window size of 30 captures a more extensive historical context. Consequently, a model trained with a window size of 30 may better comprehend longer-term trends and patterns in the data, whereas a model trained with a window size of 1 may exhibit higher sensitivity to short-term fluctuations and noise.

Two convolution layers make up the 1D convolution layer. The first layer uses 64 filters with one kernel size followed by a max pooling layer, and the second layer uses 64 filters with one kernel size followed by a max pooling layer. The returned features are then sent into the LSTM, which is subsequently transferred to the GRU layers.

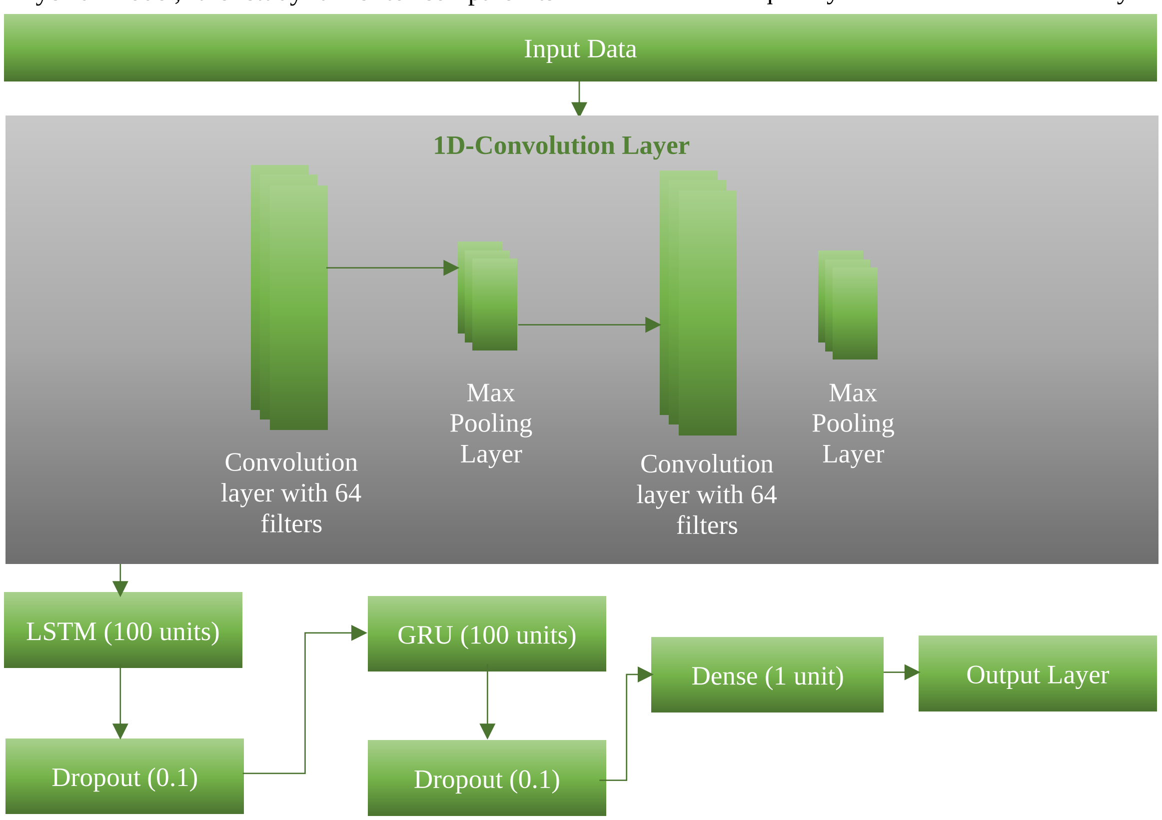

Fig. 2 CNN-LSTM-GRU architecture

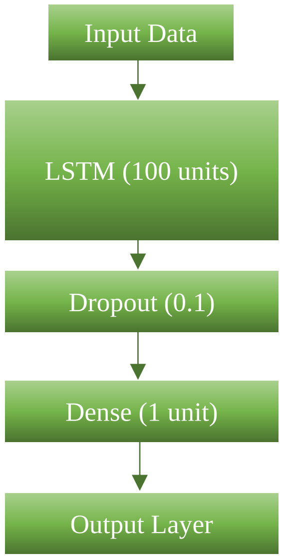

Fig. 3 LSTM architecture

To prevent overfitting, the LSTM layer is built up of 100 units and is followed by a dropout layer. The GRU layer is similarly made up of 100 units and is followed by a dropout layer. The layers' output will then be processed through a dense layer to provide the final predicted price. Max Pooling performs de-noising and dimensionality reduction while simultaneously removing all noisy activations, making it an effective noise suppressant. On the other hand, as a noise-suppression technique, average pooling just lowers dimensionality. Therefore, Max Pooling performs better than Average Pooling [16].

Based on the specified metrics, the proposed model's results will be compared to other models, LSTM and LSTM-GRU. A single layer of LSTM was used to simulate the supplied time series data for the LSTM model displayed in Figure 3. The layer was made up of 100 units and then followed by a dropout layer with a dropout rate of 0.1 to prevent overfitting.

As illustrated in Figure 4, the LSTM-GRU model architecture consists of an initial LSTM layer with 100 units, followed by a subsequent GRU layer with 100 units. To address the concern of overfitting, both layers are augmented with a dropout layer featuring a dropout rate of 0.1. The model aims to strike a balance between capturing short and long patterns in the data while avoiding excessive fitting to the training set.

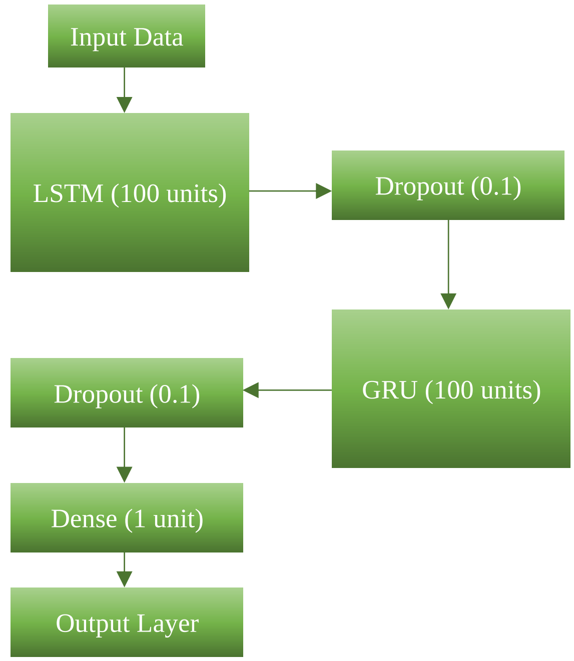

Fig. 4 LSTM-GRU architecture

3.4. Evaluation

To assess the performance of the proposed model, two commonly used metrics, namely mean absolute error (MAE) and root mean square error (RMSE), as shown in Equations 3 and 4, will be used. These metrics are widely utilized in time-series forecasting research as they provide reliable indications of how closely the model's predictions align with the actual values.

$$MAE = \frac{\sum_{i=1}^{n} | \hat{y}_i - y_i |}{n} \tag{3}$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n} (\hat{y}_i - y_i)^2}{n}} \tag{4}$$

Where ŷi is the predicted value, and yi is the original value.

The mean absolute error (MAE) measures the average absolute difference between the predicted and actual values. It provides insight into the magnitude of the errors without considering their direction. On the other hand, the root means square error (RMSE) computes the average of the squared differences between the predicted and actual values and then takes the square root of this average.

Table 3. Result of proposed model on 1-Day window size

Dataset	**TSLA**		**GOOG**		**TWTR**	
Metrics	**MAE**	**RMSE**	**MAE**	**RMSE**	**MAE**	**RMSE**
LSTM	17.790	23.252	3.053	3.772	1.122	1.665
LSTM-GRU	11.546	15.198	2.219	2.848	1.027	1.567
CNN-LSTM-GRU	10.431	13.764	2.138	2.750	1.014	1.562

Table 4. Result of proposed model on 30-Day window size

Dataset	TSLA		GOOG		TWTR	
Metrics	**MAE**	**RMSE**	**MAE**	**RMSE**	**MAE**	**RMSE**
LSTM	17.135	22.101	4.040	4.932	1.612	2.298
LSTM-GRU	16.059	20.442	3.567	4.380	1.405	2.056
CNN-LSTM-GRU	12.973	16.473	3.014	3.679	1.164	1.748

RMSE emphasizes larger errors due to the squared term, making it more sensitive to outliers. By employing these metrics, the evaluation process will provide a comprehensive understanding of how effectively the proposed model's predictions match the actual values, allowing for a thorough assessment of its performance.

4. Results and Discussion

The same feature data and model parameters were used for all the proposed models. The training process involved allocating 70% of the available data for training, 20% for testing, and 10% for validation. All the architecture is trained for a total of 30 epochs with rectified linear units (ReLU) activation function, ADAM optimizer and mean absolute error (MAE) loss function. The result of the model will then be evaluated using the MAE and root mean squared error (RMSE) using the sklearn library. The tool used for conducting this research project's code implementation is based on open-source resources. The chosen software for this task is Google Collab, and the primary programming language used in this research project is Python.

Tables 3 and 4 show the comparison of errors in the prediction of the proposed model using the specified step in the previous chapter. The results indicate that the suggested method outperforms an LSTM network in predicting stock prices for all window sizes. The difference in error between the proposed model and LSTM is significant, indicating the approach's effectiveness in predicting stock prices better than the traditional LSTM method.

The range of values in the dataset used for forecasting also plays a crucial role in determining the performance of the results. In the case of the Twitter dataset, the close price range is relatively narrow, with values ranging from 35 to 65. This could explain why the forecasting results obtained from this dataset are more accurate and have lower errors. On the other hand, the Tesla dataset has a much wider close price range, with values ranging from 200 to 400. This could explain the higher error results obtained from this dataset, as the wider range of values may have made it more difficult to predict future prices accurately.

Table 5. Average result based on TSLA, GOOG and TWTR error result on 1-Day window size

Model	MAE	RMSE
LSTM	7.322	9.563
LSTM-GRU	4.931	6.537
CNN-LSTM-GRU	4.528	6.025

Table 6. Average result based on TSLA, GOOG and TWTR error result on 30-Day window size

Model	MAE	RMSE
LSTM	7.596	9.777
LSTM-GRU	7.010	8.959
CNN-LSTM-GRU	5.717	7.300

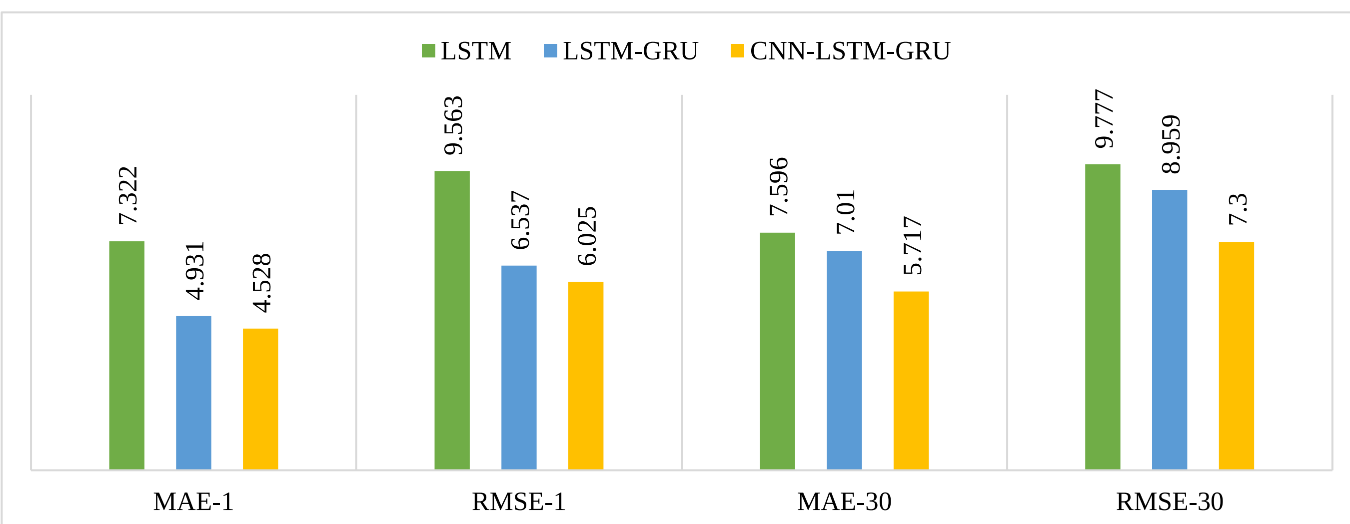

Fig. 5 Average result chart based on TSLA, GOOG and TWTR error result

To ensure fair and unbiased conclusions, an average analysis was conducted to assess the performance of each model across three datasets TSLA, GOOGL, and TWTR. The analysis considered both a 1-day and a 30-day window size, providing a comprehensive evaluation of the model's effectiveness over different window sizes. By averaging the results from Table 3 and Table 4, potential biases based on individual dataset performance were minimized. The results presented in Tables 5 and 6, along with Figure 5, are the average performance of each model in predicting stock market trends across the considered datasets.

According to the table and chart result, the LSTM model has the highest error values compared to other models, with MAE and RMSE values of 7.322 and 9.563 on 1-day window size, 7.596 and 9.777 on 30-day window size, indicating that the LSTM model alone produces unsatisfactory results.

Integrating multiple deep learning techniques, known as hybrid deep learning, is applied to enhance its performance. The research shows that using hybrid models such as LSTM-GRU outperforms the LSTM model based on Tables 5 and 6 results. However, the performance of these hybrid models can still be improved by using the CNN model as the feature extraction on the LSTM-GRU hybrid model, making it the CNN-LSTM-GRU hybrid model. The CNN-LSTM-GRU model evidences these results in Tables 5 and 6, which demonstrate the lowest error values of 4.528 MAE and 6.025 RMSE on a 1-day window size and 5.717 MAE and 7.300 RMSE on 30-day window size.

By combining CNN, LSTM and GRU, the proposed model can effectively capture both short and long-term patterns in the data. This comprehensive approach results in improved performance and more accurate predictions compared to models that solely rely on LSTM or LSTM-GRU architectures. The utilization of CNN aids in extracting relevant features and filtering out noise from the input data. The LSTM component excels at capturing long-term dependencies, while the GRU component provides faster training and convergence. Integrating these components enhances the model's capability to capture and leverage both short and long-term patterns in the data synergistically, leading to superior performance and lower error rates in predicting future values.

5. Conclusion

The stock market's growth is evident; many people are already invested in it. Forecasting is widely used in this market to anticipate whether prices will rise or fall, and those who succeed will have a substantial advantage. Finding appropriate methodologies and models for estimating stock prices is a critical and difficult issue due to their high volatility and fluctuation, which has resulted in significant risks associated with investing in stock shares.

The result of the study reveals that the CNN-LSTM-GRU model has significantly fewer errors in predictions compared to an LSTM and LSTM-GRU model. This result demonstrates the superiority of the proposed hybrid model in accurately forecasting stock prices. Overall, these results highlight the proposed technique in stock price forecasting and similar time series forecasting tasks.

To enhance the performance of the model, the utilization of popular CNN models such as AlexNet and GoogleNet in the training process is proposed. To further improve forecast results, incorporating more variable or sentiment data from sources such as social media or news is suggested as a potential avenue for future research. This can provide a more comprehensive view of the stock market, leading to more accurate predictions.

References

[1] Rob J. Hyndman, and George Athanasopoulos, *Forecasting: Principles and Practice*, OTexts, 2018. [Google Scholar] [Publisher Link]

[2] Sima Siami-Namini, Neda Tavakoli, and Akbar Siami Namin, "A Comparison of ARIMA and LSTM in Forecasting time Series," *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[3] Li Deng, and Dong Yu, "Deep Learning: Methods and Applications," *Foundations and Trends in Signal Processing*, vol. 7, no. 3–4, pp. 197–387, 2014. [CrossRef] [Google Scholar] [Publisher Link]

[4] Deepali M. Kotambkar, and Pallavi M. Wankhede, "Hybrid LSTM/GRU-based Domain Adaptation Model for Correlation Analysis to Detect Glaucoma," *SSRG International Journal of Electrical and Electronics Engineering*, vol. 10, no. 1, pp. 168-175, 2023. [CrossRef] [Publisher Link]

[5] Peter T. Yamak, Li Yujian, and Pius K. Gadosey, "A Comparison between Arima, Lstm, and Gru for Time Series Forecasting," *Proceedings of the 2019 2nd International Conference on Algorithms, Computing and Artificial Intelligence*, pp. 49–55, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[6] Mohil Maheshkumar Patel et al., "A Deep Learning-based Cryptocurrency Price Prediction Scheme for Financial Institutions," *Journal of Information Security and Applications*, vol. 55, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[7] Viswapriya Misra, "*Time Series Forecasting with Applications to Finance*," NJIT Digital Common, 2021. [Google Scholar] [Publisher Link]

[8] G. P. Dimf, P. Kumar, and K. Paul Joshua, "CNN with BI-LSTM Electricity Theft Detection based on Modified Cheetah Optimization Algorithm in Deep Learning," *SSRG International Journal of Electrical and Electronics Engineering*, vol. 10, no. 2, pp. 35-43, 2023. [CrossRef] [Publisher Link]

[9] Ioannis E. Livieris et al., "An Advanced CNN-LSTM Model for Cryptocurrency Forecasting," *Electronics*, vol. 10, no. 3, p. 287, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[10] Saul Alonso-Monsalve et al., "Convolution on Neural Networks for High-frequency Trend Prediction of Cryptocurrency Exchange Rates using Technical Indicators," *Expert Systems with Applications,* vol. 149, p. 113250, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[11] Kyung G.O, and I.F.Myung-Suck, "Evaluation of Short-Term Interval Models for Financial Time Series Forecasting," *SSRG International Journal of Industrial Engineering*, vol. 2, no. 3, pp. 5-8, 2015. [CrossRef] [Publisher Link]

[12] Wenjie Lu et al., "A CNN-LSTM-based Model to Forecast Stock Prices," *Complexity*, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[13] Andres Vidal, and Werner Kristjanpoller, "Gold Volatility Prediction using a CNN-LSTM Approach," *Expert Systems with Applications*, vol. 157, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[14] Urooj Kaimkhani, Bushra Naz, and Sanam Narejo, "Rainfall Prediction Using Time Series Nonlinear Autoregressive Neural Network," *SSRG International Journal of Computer Science and Engineering*, vol. 8, no. 1, pp. 30-38, 2021. [CrossRef] [Publisher Link]

[15] Hamed Habibi Aghdam, and Elnaz Jahani Heravi, *Guide to Convolutional Neural Networks*, *Springer Cham*, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[16] Sumit Saha, A Comprehensive Guide to Convolutional Neural Networks — the ELI5 way, 2018. [Online]. Available: https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53