Robustness Analysis on Stock Market Prediction Method

Haonian Ji*

Department of Artificial Intelligence, Southwest Jiaotong University, Chengdu, China

*Corresponding author: 2022111346@my.swjtu.edu.cn

Abstract. Stock prediction is a vital approach used to forecast stock's future prices by analyzing historical data. This technique plays a crucial role in assisting investors in making informed decisions in the stock market. There are various methods employed for stock prediction, with the most common being traditional statistical methods and machine learning methods. Traditional statistical methods involve models such as ARIMA and GARCH, while machine learning methods involve LSTM and GRU models.This research paper aims to compare the performance of SARIMA , LSTM and GRU models in stock prediction. To enhance their performance, the models will be optimized by incorporating DWT-ARIMA-XGBoost[1], LSTM-XGBoost[2] and GRU-XGBoost techniques. These methods will then be deployed to predict stock prices, with a subsequent comparison of the results obtained from the three models. After comparing those models, this paper selects four models that have high level of prediction to study their generalization. This analysis will provide valuable insights into the effectiveness and suitability of these models for stock prediction tasks. The paper evaluates these models based on their prediction accuracy, generalization ability.

Keywords: Robustness Testing, Stock Market Prediction, Hybrid Machine Learning Method.

1. Introduction

Stock price forecasting plays a vital role in predicting future stock prices by analyzing historical data. It helps investors make informed decisions and maximize returns in the stock market. Common approaches for stock forecasting include traditional statistical methods and machine learning techniques. Traditional statistical methods include Autoregressive Integrated Moving Average (ARIMA) and Generalized Autoregressive Conditional Heteroskedasticity (GARCH) models, while machine learning techniques involve Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) neural networks[4]. This paper focuses on the DWT-ARIMA-XGBoost model, LSTM-XGBoost model and GRU-XGBoost model, comparing their predictive performance. The data used comprises the closing prices of constituents of the S&P 500 index for forecasting and training purposes.

In traditional time series analysis methods, Box et al. proposed the ARIMA model in 1970 and successfully applied it to different industries. Kimoto et al. used ARIMA to predict Japanese stock prices in 1990. Hyndman et al. proposed the SARIMA model in 2005 to capture trends and seasonality in stock data. In deep learning methods, Hochreiter and Schmidhuber proposed the LSTM model in 1997. Li et al. used LSTM to predict the A-share index in 2015 and achieved good results. These two methods have played an extremely important role in stock prediction, and the optimization of these two models has also been endless. Therefore, it is very meaningful to compare the two models and to compare the models obtained after the optimization of the two models. At the same time, since xgboost has a good advantage in capturing nonlinear details, it is very meaningful to use xgboost for optimization. In 2020, the DWT-ARIMA-XGBoost model proposed by Y. Wang and Y. Guo combined DWT and ARIMA with xgboost model to achieve good prediction effect. In addition, the method of combining LSTM and XGBoost proposed by Shen, X. for time series prediction also well predicted the time series model. Therefore, it is also very valuable to compare the two optimized models.

2. Methodology

This paper mainly uses DWT, SARIMA, LSTM, GRU, XGBOOST and other methods, and applies the methods to stock prediction.

2.1. Discrete Wavelet Transform

The Discrete Wavelet Transform (DWT) is utilized as a part of the predictive model. It possesses excellent data decomposition capabilities and plays a remarkable role in separating high and low-frequency information. DWT can decompose the original time series data into approximate and detail components at different frequencies. This decomposition allows capturing different frequency components within the data, thereby revealing local features and trends. In the stock market, stock price fluctuations exhibit different frequency components, such as long-term trends and short-term oscillations. By decomposing the data into approximate and detail components, DWT can help the researchers to capture these features. DWT excels in feature extraction. By performing DWT decomposition on the data, coefficients of different frequency components can be obtained. These coefficients can serve as input features for other predictive models, such as ARIMA and XGBoost. By combining the predictions from different models, more accurate stock price forecasts can be obtained. In this study, the approximate component obtained after DWT processing will be used as input features for SARIMA, while the detail component will be used as input features for XGBoost. Based on experimental results, it has been found that sym5 exhibits higher prediction accuracy than db4 in the sp500 dataset, and sym5 also demonstrates significantly improved processing speed compared to db4. Therefore, using sym5 is a preferable choice.

2.2. Arima

Applying DWT to decompose the dataset into approximate and detail components allows for better capture of the linear features in the data. ARIMA models are used to process the approximate component, with the aim of analyzing the autocorrelation and moving average properties of the time series data to identify patterns and trends in stock prices for accurate forecasting. Utilizing ARIMA models on the approximate data produced by DWT helps focus on modeling the linear dependencies in the time series. DWT helps extract these linear elements from the raw data into the approximation component, isolating them from any nonlinear phenomena represented in the detail components. This approach leverages the strengths of DWT and ARIMA to separately handle linear and nonlinear aspects for improved time series forecasting.

The SARIMA model is an extension of the ARIMA model that takes into account the seasonal factors in a time series. Stock data exhibits evident seasonal patterns, although they may not align with daily or monthly divisions commonly found in everyday life. Therefore, replacing ARIMA with SARIMA in the DWT-ARIMA-XGBoost framework is highly reasonable.[3]

2.3. Lstm

LSTM models are capable of capturing and learning the temporal dependencies and long-term memory in sequence data. They contain internal gating mechanisms that control information flow and forgetting, which helps address issues like vanishing and exploding gradients. LSTMs can process variable length sequences without requiring fixed length inputs. LSTM models are commonly used for time series forecasting tasks, especially on data exhibiting complex sequential patterns and long-term dependencies. By training an LSTM model, the patterns and trends in the data can be learned, which enables future predictions to be made. The LSTM's ability to store and access information over long periods of time makes it well-suited for sequence tasks involving dependencies that span many time steps.[5] This allows LSTMs to potentially extract richer contextual representations compared to other models like GRUs or standard RNNs.[6]

2.4. Gru

Using the Gated Recurrent Unit (GRU) is a viable alternative to the Long Short-Term Memory (LSTM), as they are both variants of Recurrent Neural Networks (RNNs) specially designed for processing sequential data. GRUs share some similarities with LSTMs but also have differences. Compared to LSTMs, GRU models have a simpler structure, containing only an update gate and a reset gate, which reduces the number of parameters and helps alleviate the model burden, lowering the risk of overfitting. With fewer parameters, GRU models may have a lighter computational load, so in some cases their computation speed could potentially be faster than LSTMs. However, due to its increased computational complexity, the LSTM model may exhibit superior performance within specific predictive tasks. While GRUs simplify the design of LSTMs, both aim to address the vanishing gradient problem to effectively learn from sequence data, making either suitable for time series forecasting problems depending on the characteristics of the data and task.

2.5. XGBoost

XGBoost exhibits strong nonlinear fitting capabilities. The stock market possesses complex nonlinear characteristics, and traditional linear models such as ARIMA may struggle to capture these nonlinear relationships. As a nonlinear model, XGBoost can better fit the nonlinear variations in stock prices. Therefore, the XGBoost model is better suited for handling the detailed information, making it suitable for processing the high-frequency information obtained from the DWT model. By combining the XGBoost-processed high-frequency information with the low-frequency information processed by ARIMA, more superior predictive results can be achieved.

The XGBoost model is utilized to further model and correct the residuals of the LSTM model. The LSTM model may introduce some errors when predicting stock prices, especially for complex and noisy data. By employing the XGBoost model, it becomes possible to capture and learn data patterns and trends that the LSTM model might have missed, thereby enhancing the accuracy of the predictions. In the code, the prediction residuals of the LSTM model (i.e., the differences between the actual values and the LSTM predictions)[7] are calculated, and then the XGBoost model is used to model these residuals. The XGBoost model learns how to predict more accurate residual values based on the input LSTM residuals. The final prediction is obtained by adding the LSTM predictions to the XGBoost model's residual predictions.

3. Experiment

3.1. Single Model

Different models have different way to predict stock price. In this section, we will individually forecast stock data using SARIMA, LSTM, and GRU models. The training set will consist of data up to and including the year 2016, while the test set will include data from the year 2017 onwards. We will train and compare the three basic models to evaluate their performance. And **Fig. 1** provides a detailed description of the workflow for these three models.

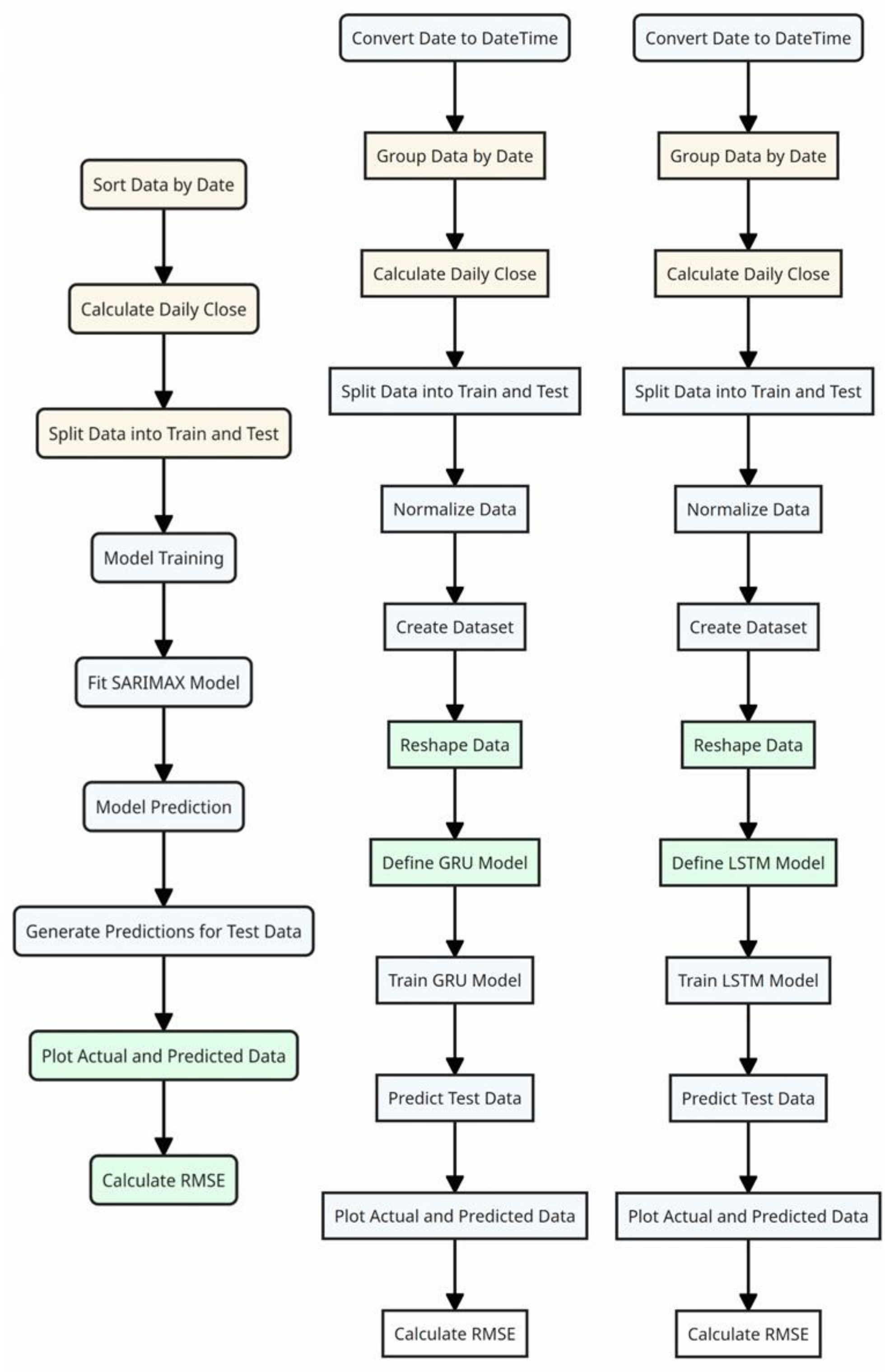


Fig. 1 From left to right are SARIMA model, LSTM model and GRU model respectively

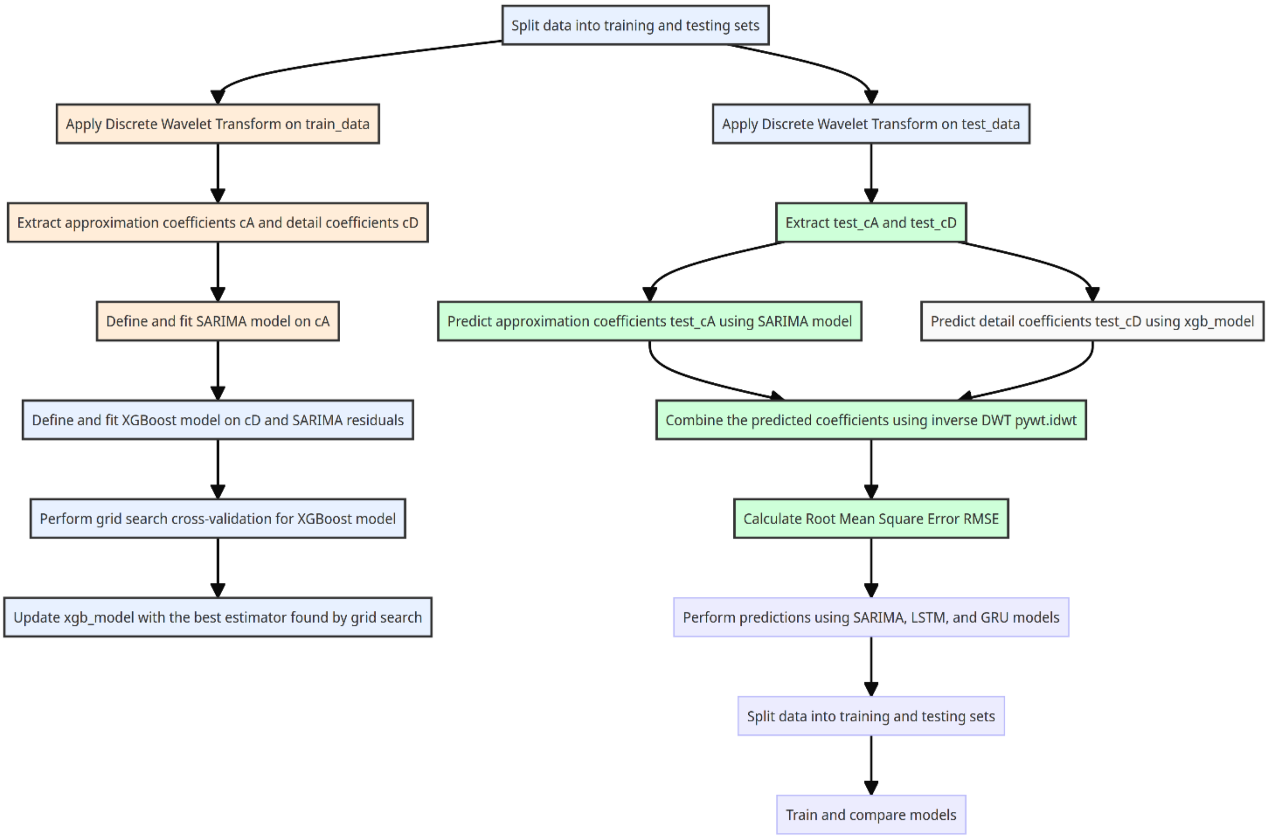


Fig. 2 DWT-SARIMA-XGBoost Model

Divide the daily_close Series into a training set (train_data) containing data up to the year 2016, and a testing set (test_data) containing data from the year 2017 onwards. Perform a discrete wavelet transform (DWT) on the train_data using the 'sym5' wavelet. The resulting approximation coefficients (cA) and detail coefficients (cD) are extracted. Define SARIMA model using the cA data, with specified order and seasonal order. Fit the SARIMA model to the cA data. Define an XGBoost model (xgb_model) with specified hyperparameters. Fit the XGBoost model using the cD data and the residual values from the SARIMA model. Define a dictionary params containing different hyperparameter values for the XGBoost model. Perform grid search cross-validation (GridSearchCV) on the XGBoost model using the cD data and SARIMA residuals to find the best hyperparameters. Update the XGBoost model (xgb_model) with the best estimator found from grid search. Perform a DWT on the test_data using the 'sym5' wavelet. Extract the approximation coefficients (test_cA) and detail coefficients (test_cD). Use the SARIMA model to predict the approximation coefficients for the test data. Use the XGBoost model to predict the detail coefficients for the test data. Combine the predicted approximation and detail coefficients using inverse DWT (pywt.idwt) to obtain the final predicted values (y_pred), as it show in **Fig. 2.** Calculate the root mean squared error (RMSE) between the actual test data (test_data) and the predicted values (y_pred). In this section, we will individually forecast stock data using SARIMA, LSTM, and GRU models. The training set will consist of data up to and including the year 2016, while the test set will include data from the year 2017 onwards. We will train and compare the three basic models to evaluate their performance.

3.2. LSTM-XGBoost

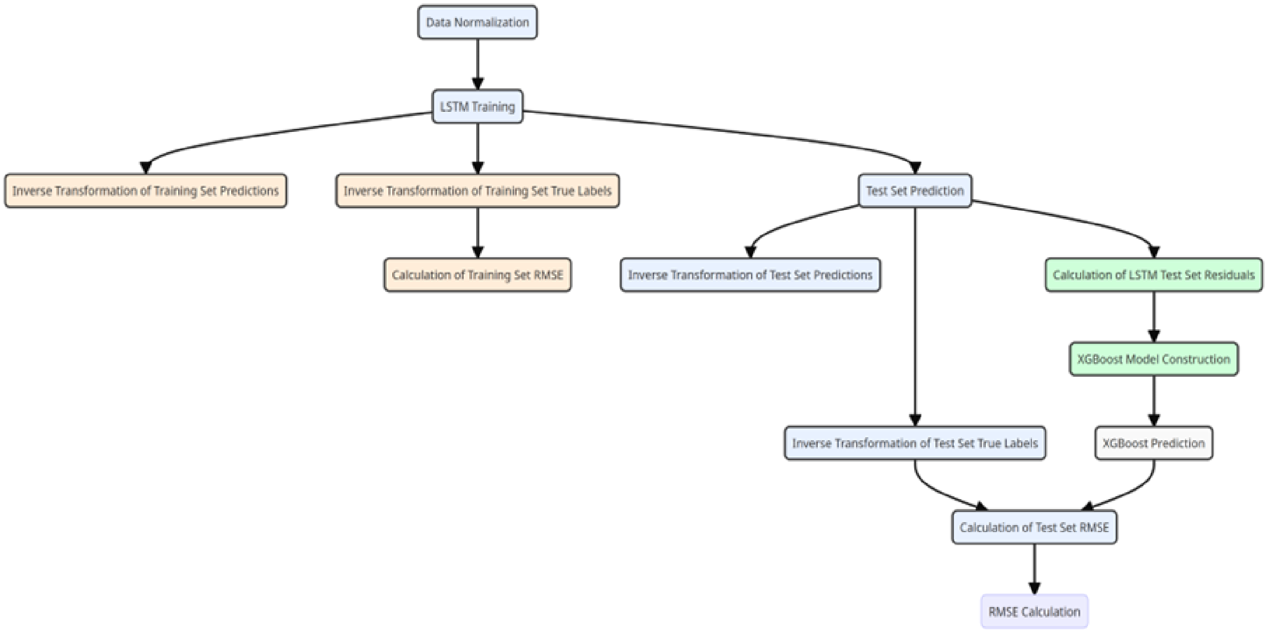


Fig. 3 LSTM-XGBoos

The process depicted in **Fig. 3** illustrates the LSTM-XGBoost .In this paper, we employed MinMaxScaler to normalize the training and testing sets, scaling the data between 0 and 1. Subsequently, the LSTM model was trained using the normalized data. After training, the model was used to predict the target variable using the training set, and the predictions were inverse-transformed to obtain the original data predictions. Similarly, the true labels of the training set were also inverse-transformed, and the root mean square error (RMSE) was calculated as a performance evaluation metric. Next, the testing set was used for prediction, and the predicted results were inverse-transformed to obtain the original data predictions. Similarly, the true labels of the testing set were inverse-transformed, and the RMSE was calculated as an assessment of the model's generalization ability on new data. Additionally, the residuals of the LSTM model on the testing set were calculated by subtracting the true labels from the LSTM predictions, providing insights into the model's prediction biases.Subsequently, an XGBoost regression model (xgb_model) was constructed and configured with the specified parameters. The flattened input features of the testing set were used as inputs, and the residuals from the LSTM model were used as the target variable for training. Then, the XGBoost model was used to predict the testing set data, yielding the final predictions. A graphical comparison between the predicted results and the true labels of the testing set was plotted to visualize the predictive performance of the model.Finally, the RMSE was calculated to assess the performance of the LSTM-XGBoost model on the testing set, specifically measuring the root mean square error between the true labels of the testing set and the predictions obtained from the LSTM model and further refined by the XGBoost model. This metric served as an evaluation of the overall predictive performance of the LSTM-XGBoost model on the testing set[9].

3.3. GRU-XGBoost

The GRU-XGBoost model is very similar to the LSTM model.Just switch LSTM into GRU.[8]

3.4. step-by-step prediction with SARIMA

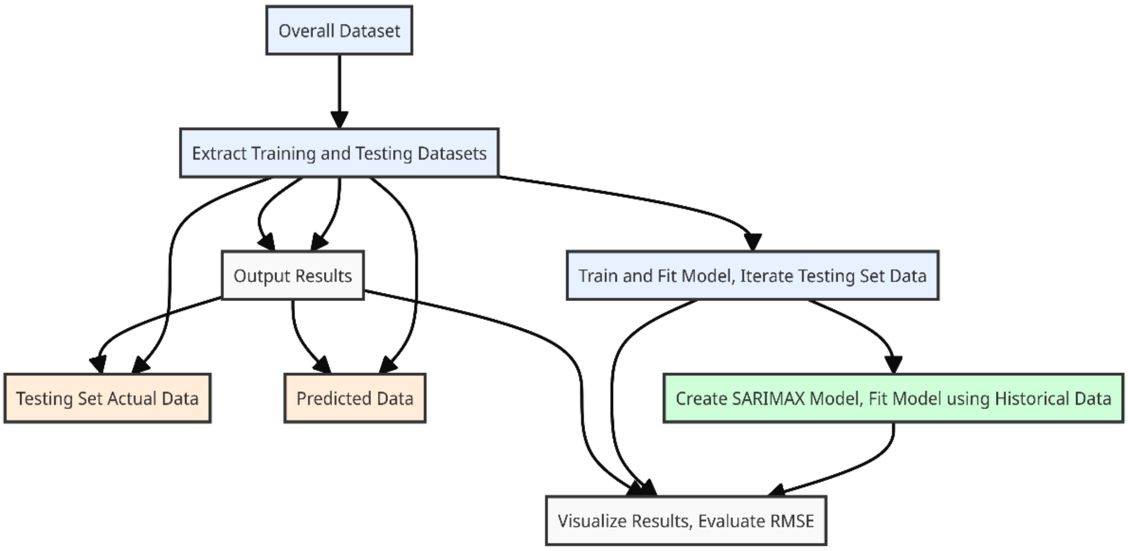


Fig. 4 step-by-step prediction with SARIMA

The step-by-step prediction with SARIMA, as shown in **Fig. 4**, is demonstrated by the following process. Extract the training and testing datasets from the overall dataset. Set the orders of the SARIMA model. Initialize a list to store the predictions. Train and fit the SARIMA model using the historical data from the training set. Iterate through the testing set, creating a SARIMAX model for each time point. Use the model to predict the next data point, append it to the predictions list, and update the historical data. Continue the iteration until the entire testing set is predicted. Output the results, including the line graph and the RMSE value.

3.5. Robustness Analysis

The study utilizes the stock data of companies listed in the S&P 500, which comprises the top 500 companies globally. The time frame of the study spans from 2005 to 2022. Pre-2015 data is used for training, while the data from 2016 is employed for prediction. The primary objective is to conduct preliminary training and prediction using various models, exploring their forecasting capabilities, and identifying the most suitable method for this dataset. The models' generalization abilities are assessed through the use of RMSE boxplots.

During the initial exploration of the models, Apple Inc.'s stock data is utilized. Multiple models are compared based on their RMSE values and prediction curves when forecasting Apple's stock. It is observed that the SARIMA model and the DWT-SARIMA-XGBoost model do not yield satisfactory results on this dataset. Consequently, these two models are not included in the subsequent analysis of generalization abilities. However, SARIMA demonstrates exceptional predictive capabilities throughout the step-by-step prediction process.

In the study on generalization abilities, the researchers select the top 50 datasets with the highest correlation from the S&P 500 dataset. Four datasets are randomly chosen from this subset for training the models. Subsequently, the trained models are utilized to predict other datasets, generating RMSE values. Four different models are trained, and their RMSE values are compared using a boxplot. This visualization method provides a more intuitive representation of the models' generalization abilities. The results show that utilizing XGBoost to learn the residuals of LSTM and GRU models significantly enhances their performance compared to the original models. Furthermore, when comparing the generalization abilities of LSTM and GRU[9], it is observed that GRU outperforms LSTM. This finding aligns with common knowledge, as the more complex gating structure of LSTM leads to poorer generalization abilities in this study.

4. Result

These instructions apply to everyone, regardless of the formatter being used

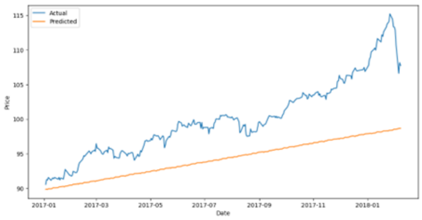
Different models have varying predictive abilities, and in order to compare their predictive capabilities, the root mean square error (RMSE) is used here to demonstrate the performance of different models. The RMSE values differ among the different models, the specific rmse is shown in the Table 1.

Table 1. Different Models' RSME

Numble	MODEL	RMSE
1	SARIMA	6.506
2	LSTM	1.590
3	GRU	1.332
4	SYM5-SARIMA-XGBOOST	2.001
5	SYM5-SARIMA-RANDOM FOREST	5.300
6	LSTM-XGBOOST	0.238
7	GRU-XGBOOST	0.234

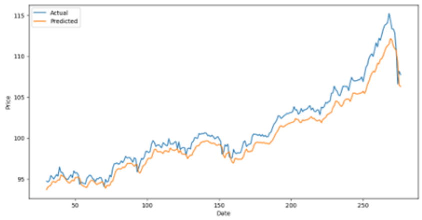
The detailed information could be showed in the figures. Just by comparing different figures, it will illustrate models' prediction ability. The study compared various models and obtained the following results. LSTM and GRU outperformed the SARIMA model in stock prediction. Additionally, using XGBoost to learn the residuals of LSTM and GRU effectively optimized the results. The DWT method indeed improved the results., which is shown in **Fig. 5**. However, replacing XGBoost with Random Forest did not yield better performance, which is shown in **Fig. 7**. It is important to note that the comparison of the two models lacks generality due to the limited number of stocks used in the analysis. The dataset is not extensive enough to capture sufficient data for SARIMA to accurately capture the overall trend, which resulted in imperfect results. Moreover, due to the characteristics of DWT, the limited amount of data hinders multiple data decompositions. Therefore, it is necessary to validate the results with a larger dataset in the future. In the case of this particular stock data, it was observed from the training loss that the DWT-SARIMA-XGBoost model is not suitable for the dataset the paper chosen. This paper also use ensemble averaging on LSTM-XGBoost and GRU-XGBoost [10], but this method made the result worse, which shows in **Fig. 8**. On the other hand, GRU-XGBoost showed promising results for stock prediction, Increasing the number of epochs for GRU and LSTM did not improve the prediction results; in fact, it led to worse performance.

Speaking of generalization ability, the residual learning method explored in this paper has played a significant role in improving generalization ability. Both LSTM and GRU methods have benefited from the inclusion of residual learning through XGBoost Fig. 6. Moreover, the prediction results also indicate that LSTM, due to its complex gate structure, exhibits weaker generalization ability compared to GRU. **Fig. 9** shows this very well.

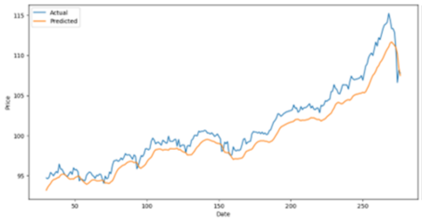


A

B

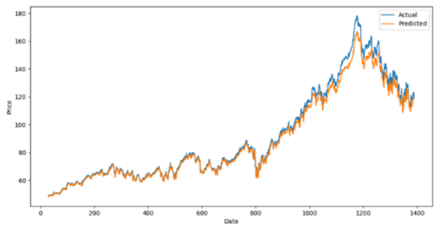


C

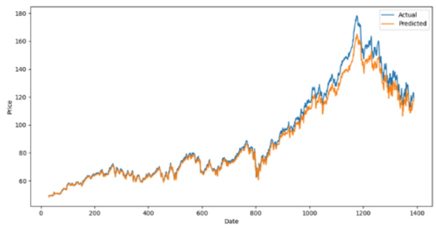


D

Fig. 5 A: The table represents the SARIMA model. B: The table represents the DWT-SARIMA-XGBoost model. C: The table represents the LSTM model. D: The table represents the GRU model

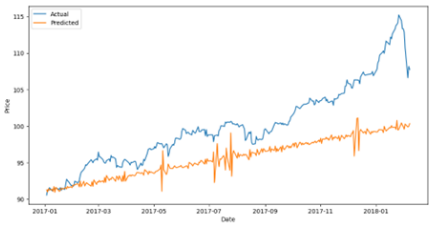


A



B

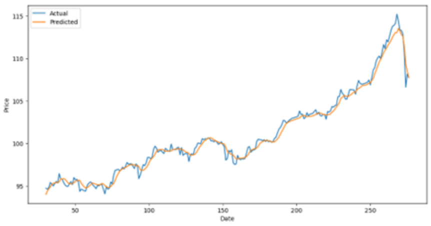
Fig. 6 A: The table represents the LSTM-XGBoost model. B: The table represents the GRU-XGBoost model.



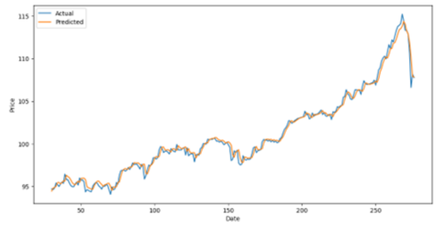
A

B

Fig. 7 A: The table represents the DWT-SARIMA-XGBOOST model. B: The table represents the DWT-SARIMA-RANDOM FOREST WITH XGBOOST model.



A



B

Fig. 8 A: The table represents the LSTM-XGBOOST PLUS Model. B: The table represents GRU-XGBOOST PLUS Model

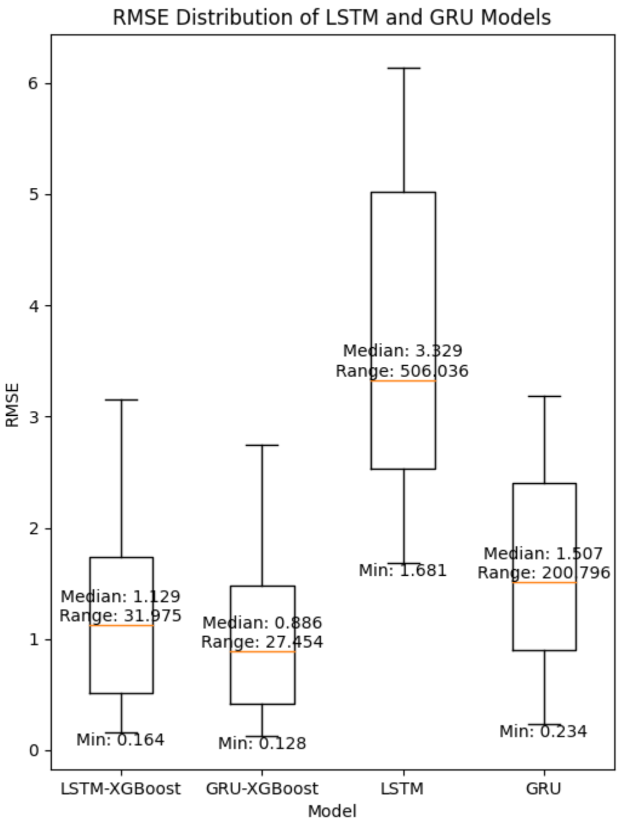


Fig. 9 This box plot code represents the box plots of the mean squared error (MSE) distribution during the testing process after training for four different models: LSTM model, GRU model, LSTM-XGBoost model and GRU-XGBoost model.

5. Conclusion

This paper compare DWT-SARIMA-XGBoost , LSTM-Boost and GRU-Boost models, shows that DWT-SARIMA-XGBoost model have better performance than SARIMA model and LSTM-XGBoost GRU-XGBoost models do well in the sp500 dataset when compare with DWT-SARIMA-XGBoost. Then this paper also compare LSTM-XGBoost and GRU-XGBoost models' generalization ability. To better demonstrate the results of the models' generalization abilities, this paper uses box plots to compare the distribution of RMSE values across different models. Additionally, LSTM and GRU models are included as references to highlight the role of XGBoost in the model. From the two sets of comparisons between LSTM and GRU, and LSTM-XGBoost and GRU-XGBoost, it can be observed that GRU exhibits superior generalization abilities compared to LSTM.However, it should be noted that due to the limited availability of data in this study, there may be some biases in the selection and evaluation of the data. Therefore, it is inevitable that there might be some omissions. Furthermore, the research on the generalization abilities of the SARIMA model in the context of stepwise prediction is lacking. It is hoped that future studies can delve into and optimize the usage of SARIMA on this dataset, with a more detailed examination of its generalization capabilities.

References

[1] Y. Wang and Y. Guo, "Forecasting method of stock market volatility in time series data based on mixed model of ARIMA and XGBoost," China Communications, vol. 17, no. 3, pp. 205-221, 2020. DOI:10.23919/JCC.2020.03.017.

[2] Hoang Pham, Trinh Vuong, Dat Tan, Khoi Tieu, Pham Mai, Hoang Uyen, The Pham, and Bao. "Stock-Price Forecasting Based on XGBoost and LSTM," Computer Systems Science and Engineering, vol. 40, no. 1, pp. 237-246, 2022.

[3] A. Kanavos, F. Kounelis, L. Iliadis, et al. "Deep learning models for forecasting aviation demand time series," Neural Computing and Applications, vol. 33, pp. 16329-16343, 2021. DOI: 10.1007/s00521-021-06232-y.

[4] I.R. Parray, S.S. Khurana, M. Kumar, et al. "Time series data analysis of stock price movement using machine learning techniques," Soft Computing, vol. 24, pp. 16509-16517, 2020. DOI: 10.1007/s00500-020-04946-3.

[5] G. Vargas, L. Silvestre, L. Rigo Júnior, and H. Rocha, "B3 Stock Price Prediction Using LSTM Neural Networks and Sentiment Analysis," IEEE Latin America Transactions, vol. 20, no. 7, pp. 1067-1074, July 2022. DOI: 10.1109/TLA.2021.9827469.

[6] L. Tian, L. Feng, L. Yang et al. "Stock price prediction based on LSTM and LightGBM hybrid model," Journal of Supercomputing, vol. 78, pp. 11768-11793, 2022. DOI: 10.1007/s11227-022-04326-5.

[7] Xijun Zhang and Qirui Zhang, "Short-Term Traffic Flow Prediction Based on LSTM-XGBoost Combination Model," Engineering and Science in Computer Modeling, 2020, 10: 95-109.

[8] H. Niu and K. Xu, "A hybrid model combining variational mode decomposition and an attention-GRU network for stock price index forecasting," Mathematical Biosciences and Engineering, vol. 17, no. 6, pp. 7151-7166, 2020. DOI: 10.3934/mbe.2020367.

[9] Y. Touzani and K. Douzi, "An LSTM and GRU based trading strategy adapted to the Moroccan market," Journal of Big Data, vol. 8, no. 126, 2021. DOI: 10.1186/s40537-021-00512-z.

[10] G. R. Patra and M. N. Mohanty, "An LSTM-GRU based hybrid framework for secured stock price prediction," Journal of Statistics and Management Systems, vol. 25, no. 6, pp. 1491-1499, 2022. DOI: 10.1080/09720510.2022.2092263.