Vietnam's Stock Market Prediction using statistical algorithm, machine learning and deep learning model

1st Tran Hoang Nhat
IS403.O22.HTCL
University of Information Technology
Ho Chi Minh City
21522420@gm.uit.edu.vn

2nd Thai Ngoc Dung
IS403.O22.HTCL
University of Information Technology
Ho Chi Minh City
21521982@gm.uit.edu.vn

3rd Nguyen Hoang Vu
IS403.O22.HTCL
University of Information Technology
Ho Chi Minh City
21522799@gm.uit.edu.vn

ABSTRACT

Stock prediction plays a crucial role in economic planning and investment decision-making. This study focuses on leveraging advanced statistical and machine learning techniques to forecast stock prices in the Vietnamese market, with a particular emphasis on Vietnam Airlines JSC (HVN), SCSC Cargo Service Corporation (SCS), and Vietjet Aviation Joint Stock Company (VJC). Eight time-series models, including Linear Regression, ARIMA, RNN, GRU, LSTM, SEMOS, Stacking Model, and FCN, are utilized for prediction. Evaluation of these models is conducted using metrics such as MAPE, RMSE, and MSLE. Additionally, related research on stock price prediction employing SEMOS, Stacking, and FCN algorithms is reviewed, showcasing their effectiveness in other markets. The dataset spans from 01/03/2019 to 29/02/2024, and only the "Close" prices (VND) are considered for analysis. This study aims to provide stakeholders with actionable insights for informed decision-making in the Vietnamese stock market.

I. INTRODUCTION

Stocks represent a vital component of Vietnam's economic landscape, serving as a cornerstone for economic development and fostering investor participation. The stock market not only facilitates capital infusion into enterprises, enabling their expansion and innovation but also serves as a lucrative avenue for investors to seek returns on their investments.

Recognizing the pivotal role of stocks in driving economic progress, our endeavor focuses on leveraging advanced statistical and machine learning techniques to predict stock prices, thereby aiding decision-making processes within the realm of investments and economic planning.

There are numerous amounts of time-series models, this project uses eight time-series models to predict stock price in Vietnam: Linear regression, ARIMA, RNN, GRU, LSTM, SEMOS, Stacking Model, FCN.

Evaluation of predictive models is conducted through a comprehensive analysis based on multiple criteria, including Mean Absolute Percentage Error (MAPE), Root Mean Square Error (RMSE), Mean Squared Logarithmic Error (MSLE), and result of data division methods. Through this meticulous evaluation process, we determine which one is best for estimating the price. Thereby providing stakeholders with actionable insights and informed decision-making tools.

II. RELATED WORKS

Due to the profitability and importance of stocks, many methods have been developed to predict stock prices. Here, we will utilize 8 algorithms: Stacking Model, FCN, SEMOS, Linear regression, ARIMA, RNN, GRU and LSTM. Below are some related research papers on stock price prediction using these 8 algorithms.

In the study by Philip Ngare,Dennis Ikpe and Samuel Asante Gyamerah, three models were employed: AdaBoost, KNN, and Stacking (AdaBoost and KNN serve as base-level classifiers, and GBM acts as the meta-level classifier). They utilized a dataset obtained from the Nairobi Stock Exchange, and the results showed that Stacking outperformed the others with an accuracy of 0.7810, an area under the curve of 0.8238, a kappa of 0.5516, and an out-of-bag error rate (OOB) of 21.89% [1].

In the research by Shima Nabiee and Nader Bagherzadeh, they compared the performance of 6 different models, including FCN, SegNet, U-Net, DeepLab V3+, and two proposed models with 20 and 40 days input frames, respectively. The results showed that when using price frames as input, FCN achieved an AUC of 0.79 and an Accuracy of 70.25%. However, for trend input, the FCN model showed the best results with an AUC of 0.77 and an accuracy of 70.77%. They

also found that when the input frame is 1 (40 days), it gives better short-term prediction results and when the number of input frames increases, the long-term results are improved [2].

The research of David Jobst, Annette Möller, and Jürgen Groß has shown that SEMOS is used to enhance the forecasting performance of numerical weather prediction (NWP) models by employing finite Fourier series, making it a clearly seasonal and trend-aware time series model. The results show that SEMOS provides more accurate weather forecasts based on evaluation metrics such as CRPS, LogS, and RMSE. Additionally, SEMOS maintains good forecasting performance across different time horizons. These advantages indicate that SEMOS could be a good choice for applying to stock market forecasting[3].

In Xiwen Jin and Chaoran Yi's study, they found that GRU yielded the best results, followed by LSTM and Linear Regression. The R2 scores were as follows: LSTM 0.84, GRU 0.86, Linear Regression 0.73, and the MSE scores were: LSTM 7.06, GRU 6.26, Linear Regression 6.64 [4]. In another research by Dias Satria, four models were employed: ARIMA, RNN, LSTM, and GRU. The results showed that GRU exhibited the best performance in predicting stock prices, while ARIMA was deemed unsuitable due to the nonlinear characteristics of the data, violating the assumption of white noise in the estimation of ARIMA Box-Jenkins parameters [5].

III. MATERIALS

A. DATASET

The historical stock price of Vietnam Airlines JSC (HVN), SCSC Cargo Service Corporation (SCS) and Vietjet Aviation Joint Stock Company (VJC) from 01/03/2019 to 29/02/2024 will be applied. The data contains column such as Date, Open, High, Low, Close, Volume. As the goal is to forecast close prices, only data relating to column "close" (VND) will be processed.

B. DESCRIPTIVE STATISTICS

TABLE 1. HVN, SCS, VJC's Descriptive Statistics

	HVN	SCS	VJC
Count	1,245	1,253	1,252
Mean	20,266	63,678	117,882
Std	6,462	8,429	14,232
Min	8,610	37,320	93,800
25%	13,500	59,500	105,500
50%	21,011	64,629	117,400
75%	25,000	67,350	129,000
Max	34,753	85,420	149,000

FIGURE 1. HVN stock price's boxplot

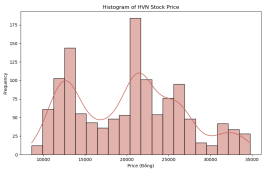

FIGURE 2. HVN stock price's histogram

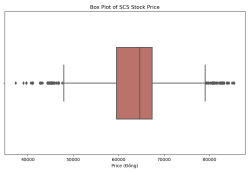

FIGURE 3. SCS stock price's boxplot

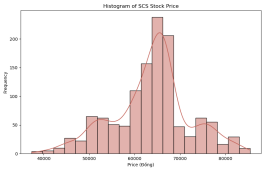

FIGURE 4. SCS stock price's histogram

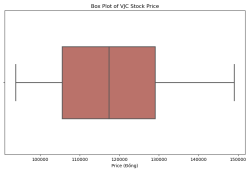

FIGURE 5. VJC stock price's boxplot

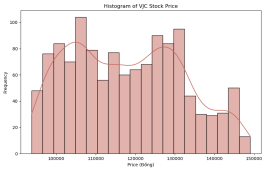

FIGURE 6. VJC stock price's histogram

IV. METHODOLOGY

A. LINEAR REGRESSION

Linear Regression is a method of statistical analysis used to determine the relationship between a dependent variable and one or many independent variables. It is used to predict the value of the dependent variable based on the value of the independent variables.

A multiple linear regression model has the form:

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_k X_k + \varepsilon$$

Where:

- Y is the dependent variable.
- $X_1, X_2, \ldots, X_k$ are the independent variables.
- β_0 is the intercept term.
- $\beta_1, ..., \beta_k$ are the regression coefficients for the independent variables.
- ε is the error term.

B. ARIMA

ARIMA(Autoregressive Integrated Moving Average) is a statistical forecasting method widely used in time series analysis. This model incorporates autoregressive (AR), moving average (MA) and Integrated (I) components to capture the relationship between current and past values of a time series. Auto Regression (AR):

$$y_t = \mu + \sum_{i=1}^{p} \gamma_i y_{t-i} + \epsilon_t \quad [6]$$

In there : y_t is the current value; μ is the constant term; p is the number of autoregressive terms; γ_i is the autocorrelation coefficient and ε_t is error.
Moving Average (MA):

$$y_t = \mu + \sum_{i=1}^{q} \theta_i \epsilon_{t-i} + \epsilon_t \quad [6]$$

In there : y_t is the current value; μ is the constant term; q is the number of terms in the moving average; θ_i is the moving average coefficient and ε_t is error.
Integrated (I):

$$\mathrm{I}(d=0) : \Delta y_t = y_t$$

$$\mathrm{I}(d=1) : \Delta y_t = y_t - y_{t-1}$$

$$\mathrm{I}(d=2) : \Delta\left(\Delta y_t\right) = (y_t - y_{t-1}) - (y_{t-1} - y_{t-2})$$

In there : d is the number of differences required to make it a stationary sequence

After combining them, we will have the ARIMA (p, d, q) express as follow:

$$y_t = \mu + \sum_{i=1}^{p} \gamma_i y_{t-i} + \sum_{i=1}^{q} \theta_i \epsilon_{t-i} + \epsilon_t \quad [6]$$

C. SEMOS

SEMOS (Smooth EMOS) is a statistical approach used for post-processing ensemble forecasts, particularly to handle seasonal variations in the predictive distribution parameters. SEMOS integrates seasonal effects directly into the estimation of location and scale parameters, enhancing the accuracy of ensemble forecasts, especially for quantities exhibiting seasonal variability.

The first parameter is location, in general, represents the central tendency or the "location" of the predictive distribution.In SEMOS, the location parameter is modeled as a function of the ensemble mean forecast and seasonal effects in order to capture systematic biases or trends in the ensemble forecasts:

$$\mu_S(t) = a_0 + f_0(t) + (a_1 + f_1(t)).x(t)$$

[7] Where:

- $\mu_S(t)$ is the location parameter at time t.
- a_0, a_1 are coefficients representing the baseline intercept and slope.
- $f_0(t), f_1(t)$ are seasonal effects modeled using cyclic regression splines.
- $x(t)$ is the ensemble mean forecast at time t.

The scale parameter represents the spread or "scale" of the predictive distribution.It accounts for variations in forecast uncertainty over time, including factors such as model errors and ensemble spread, modeled as a function of the empirical ensemble standard deviation and seasonal effects.

$$log(\sigma_S(t)) = b_0 + g_0(t) + (b_1 + g_1(t)).s(t)$$

[7] Where:

- $\sigma_S(t)$ is the scale parameter at time t.
- b_0, b_1 are coefficients representing the baseline intercept and slope.
- $g_0(t), g_1(t)$ are seasonal effects modeled using cyclic regression splines.
- $s(t)$ is the empirical ensemble standard deviation at time t.

D. STACKING

Stacking, short for Stacked Generalization, is a machine learning algorithm belonging to the Ensemble Learning category. A basic Stacking model is usually divided into two levels: level-0 models and the level-1 model. Level-0 models (base-models) are the foundational models that learn directly from the dataset and produce predictions for the level-1 model (meta-model). The meta-model is trained based on the predicted outputs of the base models. These outputs, combined with the labels, form the input-output pairs during the training process of the meta-model. In this work, ARIMAX and RNN are used as base-models and Linear Regression is selected as meta-model.

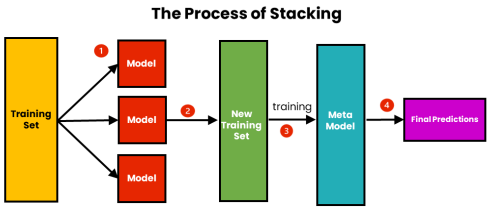

FIGURE 7. Stacking Architecture [8]

REFERENCES

[1] Samuel Asante Gyamerah,Philip Ngare,Dennis Ikpe. ”On Stock Market Movement Prediction Via Stacking Ensemble Learning Method” , 04-05 May 2019, doi: 10.1109/CIFEr.2019.8759062

[2] Shima Nabiee,Nader Bagherzadeh. "Stock Trend Prediction: A Semantic Segmentation Approach", 9 Mar 2023, doi: https://doi.org/10.48550/arXiv.2303.09323

[3] David Jobst, Annette Möller, Jürgen Groß. "Time Series based Ensemble Model Output Statistics for Temperature Forecasts Postprocessing", 1 Feb 2024, doi: https://doi.org/10.48550/arXiv.2402.00555

[4] X. Jin and C. Yi, “The Comparison of Stock Price Prediction Based on Linear Regression Model and Machine Learning Scenarios,” presented at the 2022 International Conference on Bigdata Blockchain and Economy Management (ICBBEM 2022), Atlantis Press, Dec. 2022, pp. 837–842. DOI: 10.2991/978-94-6463-030-5_82.

[5] D. Satria, “PREDICTING BANKING STOCK PRICES USING RNN, LSTM, AND GRU APPROACH” Appl. Comput. Sci., vol. 19, pp. 82–94, Mar. 2023, DOI: 10.35784/acs-2023-06.

[6] S. Kumar, A. Gupta, K. Arora, and K. Vatta, “Effect of Rainfall in Predicting Tomato Prices in India: An Application of SARIMAX and NARX Model” vol. 32, pp. 159–164, Dec. 2022.

[7] David Jobst, Annette Möller, Jürgen Groß, "Time Series based Ensemble Model Output Statistics for Temperature Forecasts Postprocessing", Feb. 2024, doi:https://doi.org/10.48550/arXiv.2402.00555

[8] Brijesh Soni, "Stacking to Improve Model Performance: A Comprehensive Guide on Ensemble Learning in Python", doi: