International Journal of Applied Sciences: Current and Future Research Trends
(IJASCFRT)
ISSN: 2790-3990
https://ijascfrtjournal.isrra.org/index.php/Applied_Sciences_Journal

Forecasting ACB Stock Prices using Machine Learning Models and Vietnamese News Sentiment Analysis

Luong Nhu Y[a], Bui Hai Ly[b], Ta Ngoc Minh[c], Ta Quang Chieu[d*]

[a]*Faculty of Mathematics and Computer Science, University of Science, Ho Chi Minh City, Vietnam*
[b]*VCREME Center of Research, Ho Chi Minh City, Vietnam*
[c]*Department of Economics and Management, Vietnamese-German University, Vietnam*
[d]*Faculty of Computer Science and Engineering, Thuyloi University, Hanoi 10000, Vietnam*
[a]*Email: 22C01029@student.hcmus.edu.vn*, [b]*Email: ly.bui@mservice.com.vn*
[c]*Email: 14316@student.vgu.edu.vn*
[d]*Email: quangchieu.ta@tlu.edu.vn*

Abstract

This paper presents a study on forecasting the stock close price of ACB bank from 2012 to 2022 using various learning machine models. The models used in this study include Decision Tree, Random Forest, and LSTM, which are combined with sentiment analysis for Vietnamese news using the Pho Bert approach. To evaluate the performance of the models, R2 and RMSE are employed as evaluation metrics. The results indicate that the LSTM model with news sentiment analysis provides the best performance in both evaluation metrics. This study contributes to the understanding of the effectiveness of combining machine learning models with sentiment analysis for forecasting stock prices.

Keywords: Decision Tree; Deep learning; LSTM; Random Forest; Sentiment analysis.

1. Introduction

The stock market is an ever-changing and unpredictable system that can be influenced by various factors, including the sentiment expressed in financial news articles. As a result, traders and investors often rely on stock price forecasting techniques to make informed investment decisions. In recent years, sentiment analysis and deep learning models have become popular tools for forecasting stock prices.

In this paper, we aim to forecast the stock price of the Asia Commercial Joint Stock Bank (ACB) from the period of 2012 to 2022, using sentiment analysis combined with a deep learning model.

Received: 4/1/2023
Accepted: 4/30/2023
Published: 5/17/2023

----

* Corresponding author.

Although many studies have explored this issue, only a few have used the Vietnamese language for sentiment analysis. To overcome this limitation, we utilise the PhoBert model, which is specifically designed for Vietnamese natural language processing, to analyse Vietnamese financial articles.

To build the deep learning model, we use three different algorithms: Decision Tree, Random Forest, and Long Short-Term Memory (LSTM). These algorithms have been widely used in finance and have shown promising results in stock price forecasting. By combining sentiment analysis and deep learning techniques, we aim to provide a more accurate and reliable forecast of ACB's stock price.

The significance of this study lies in its contribution to the limited research on Vietnamese sentiment analysis for stock price forecasting. Moreover, the use of deep learning models is relatively new in this field, and the comparison of different algorithms can provide insights into the most effective method for forecasting stock prices. The results of this study can be valuable for traders, investors, and financial institutions who seek to make informed decisions about ACB's stock.

The present paper is structured as follows: Section 2 provides a comprehensive review of the relevant literature, while Section 3 expounds on the models under consideration. Section 4 presents the empirical findings and results obtained from the models. Finally, Section 5 delineates the conclusions and furnishes a discussion of further avenues of inquiry.

2. Literature review

2.1. Models for Predicting Stock Prices

Research methods that prove or disprove the existence of the Efficient Market Hypothesis are divided into three categories based on the choice of variables and the technical method. The first method, simple regression techniques on cross-sectional data, was introduced in the early days of econometrics [1, 2, 3]. The second method consists of studies applying time series models such as Autoregressive Integrated Moving Average (ARIMA), Granger Causality Test, Autoregressive Distributed Lag (ARDL), and Quantile Regression to forecast stock prices [4, 5].

With technological advances in the past decade, machine learning techniques have been widely used to increase the accuracy of forecasting stock trends. Many deep learning models, such as Random Forest, Decision Tree, CNN, LSTM, et cetera, are currently used for stock price forecasting [6].

2.2. Text sentiment analysis for news

News is an essential data source for fundamental stock analysis in all markets worldwide. Therefore, applying this data source to stock price forecasting models has been engaging for a long time, especially in natural language processing applications. Sentiment analysis based on word vectors is the preferred method of most stock price forecasting studies applying machine learning and news processing.

Li and his colleagues (2020) tokenized the entire text, returned it to word vectors, and mapped it with sentiment dictionaries [7]. Research shows that dictionaries focusing on financial topics increase accuracy by 120% compared to other dictionaries. They also said they would improve the model by extracting events that could affect stock prices instead of processing all the news.

Dey et.al (2018) used word and paragraph vectors to classify news according to six PESTEL factors based on Distributed Memory Model [8]. The research proposed a two-layer LSTM mode.

Rahman and his colleagues (2017) used N-gram approaches, including Unigram, Bigram, and Trigram, by tokenizing words [9]. Bigram gives the results that best represent the message of the news.

Jaggi and his colleagues (2021) performed experiments on different combinations of built-in models, such as BERT, FinBERT, FinALBERT, Word2Vec, Fasttext, FinALBERT, and other time series forecasting models [10]. Research shows the best results when using labeled sentiment rating (positive, negative); BERT is the best model.

Bi (2022) applies a classical machine learning algorithm for word segmentation: Hidden Markov Model, the Viterbi algorithm for a decoding problem, and BI-LSTM for sentiment classification [11].

2.3. Studies on Predicting Stock Prices in Vietnam's Financial Market

One of the first studies to apply text mining techniques to VN-Index price prediction came from Thanh and his colleagues (2014), with a dataset for the period 2012-2012 [12]. The related English news articles were collected from the http://indochinastock.vn website, http://vietnamnews.vn website and categorized into Positive, Negative, and Neutral by several machine learning algorithms (ANN, kNN, Naive Bayes, and SVM). The results show that SVMs performance was better than other classifiers. After that, the accuracy of predicting the daily movement of the VN-index was improved by combining Linear Support Vector Machine Weight and the Support Vector Machine algorithm.

Tran and his colleagues (2021) applied PhoBERT to analyze Vietnamese news text to classify emotions into negative, positive, and neutral [13]. After that, the news score combined with the closing price of FPT stock is entered as the input data of the LSTM and LSTM-attention models to predict the closing price. The results show that the LSTM-attention model combined with news gives the best results. In addition, the study also provides evidence that incorporating news variables will provide better forecasting accuracy than using historical prices alone.

Le and his colleagues (2022) also applied the mining technique combined with machine learning models (Decision Trees, Random Forest, KNNs, and SVM) to forecast VNIndex over a long period from 2001 to 2021 [14]. The news related to the financial market in Vietnam is taken from 4 different reputable Vietnamese newspapers and labeled as increasing, decreasing, and staying the same based on the price change of the equivalent day. The results show that the SVM model gives the best VNindex forecast results when combined with information from the Vietstock newspaper.

Moreover, the study by Tran and his colleagues (2011) focuses on forecasting the stock price of FPT Group using PhoBert to classify news-based emotions as negative, neutral, or positive [15]. They proposed a NEU-Stock model that uses the LSTM-Attention model with historical closed prices and the impact of news as variables to predict the stock price of the following day. After training the model on a dataset comprising 1800 days of FPT stock price data, their model achieved the best results with an RMSE error of 730.754 and a coefficient determinant R2 up to 0.933.

3. Theoretical background

3.1. News Processing (Sentiment Scoring) using PhoBERT

PhoBERT is a pre-trained language model that utilises the transformer architecture and is specifically designed for natural language processing (NLP) tasks in the Vietnamese language [16]. PhoBERT employs Facebook's RoBERTa-like approach and architecture, which was introduced by Facebook in mid-2019 as an improvement over the previous BERT model [17]. RoBERTa introduces several changes to the BERT model, such as longer pre-training, dynamic masking, and more training data, which have been shown to improve the model's performance on downstream NLP tasks.

During the fine-tuning process, the model learns to map the input text to a probability distribution over the positive and negative sentiment labels, with higher probabilities indicating the predicted sentiment.

PhoBERT's training is based on fine-tuning RoBERTa on a large corpus of Vietnamese text data, and has achieved state-of-the-art performance on a range of NLP tasks in Vietnamese (Nguyen and his colleagues 2020). The use of pre-training and fine-tuning techniques have been shown to be effective in achieving high performance in NLP tasks across different languages, including Vietnamese.

3.2. Decision Tree

The Decision Tree technique is a prevalent method of supervised learning that can be used for solving problems in both regression and classification [18]. The aim of this approach is to predict a specific target by creating simple decision rules based on the available dataset and its corresponding features. Utilizing this model offers two benefits: interpretability and versatility in solving problems with diverse outputs. However, the potential drawback is the creation of overly complex decision trees that result in overfitting.

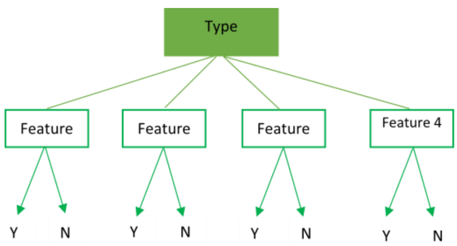

Figure 1: A schematic illustration of Decision Tree [19].

The decision tree technique belongs to the category of time series prediction models and operates as an inductive learning algorithm that learns from examples. Additionally, it can classify a set of cases that are unordered and uncontrolled, and it employs a top-down recursive approach to represent them as a "tree".

Furthermore, internal nodes of the decision tree are used for allocation selection, and the decision tree is pruned. The decision tree algorithm is characterized by its simplicity in description, rapid modeling speed, and the ease of interpretation of prediction results. The current study employs the C5.0 algorithm for decision tree classification, which is an improved step-by-step development from the original ID3 algorithm to the C4.5 algorithm. After the implementation of the proposed enhancements, the overall performance of the algorithm significantly improves. Typically, the decision tree's growth and division process primarily hinge on two aspects: selecting the best split variable from a multitude of input variables and identifying the optimal split point from numerous values of the current split variable. In terms of attribute selection for splitting, the C5.0 decision tree employs the "gain ratio" as the split attribute for the current node, which evaluates the data's breadth and uniformity. Let be a training set comprising samples that consist of m distinct classes (i=1, 2, ..., m), the number of samples of each class is , D is an attribute of the training sample set , which has k different values. Based on these values, can be partitioned into k distinct subsets. represents the i-th subset (i=1, 2, ..., k), and represents the number of samples in the subset , then the information gain Gain(S, D) can be expressed as:

$$Gain(S, D) = I(s_1, s_2, \dots, s_m) - E(S, D) \qquad (1)$$

where:

$$I(s_1, s_2, \dots, s_m) = -\sum_{i=1}^{m} p(x_i) \, log_2(x_i) \qquad (2)$$

which is the entropy of the sample set, $p(x_i)$ represents the probability of each category and $\sum_{i=1}^{m} p(x_i) = 1$, $E(S, D)$ represents the weighted sum of entropy of k subsets divided by attribute D. This split information item can be expressed as:

$$Split\_Info(S, D) = \sum_{i=1}^{k} \{(|S_i|/s) \, log_2(|S_i|/s)\} \qquad (3)$$

This split information item is called the entropy of the data set S on the attribute D. The more uniform the value distribution of the sample on the attribute D, the larger the value of this split information item. The gain ratio can be expressed as:

$$Gain\_Ratio(S, D) = \frac{Gain(S,D)}{Split\_Info(S,D)} \qquad (4)$$

Evidently, as the breadth of the attribute D increases, the data set's uniformity strengthens, resulting in a higher split information item but a lower gain ratio. The subsequent step involves selecting the optimal split point. If the best split attribute is a discrete variable with K categories, the sample set is divided into k branches or groups, forming the decision tree's k branches. In instances where the optimal split attribute is a continuous variable, the MDLP binning technique is used to determine the minimum group limit. Samples below the threshold value are grouped together, while those exceeding the threshold are placed in a separate group, resulting in a decision tree that comprises two branches.

3.3. Random Forest

Decision trees can be used for various machine learning applications [20]. The Decision tree Model that are extensively grown to learn intricate patterns tend to overfit the training sets, making them highly sensitive to minor data fluctuations that cause the tree to diverge. This phenomenon arises due to decision trees' low bias and high variance. Random Forests resolve this issue by training numerous decision trees on various feature space subspaces, albeit with a slightly higher bias.

A Random Forest model comprises a considerable number of decision trees that generate forecast results, which are averaged to form a forest. The algorithm integrates three random concepts: random selection of training data during tree formation, random selection of variable subsets when dividing nodes, and using only a portion of all variables for splitting each node in the basic decision tree. During the Random Forest training process, each decision tree learns from a random dataset sample.

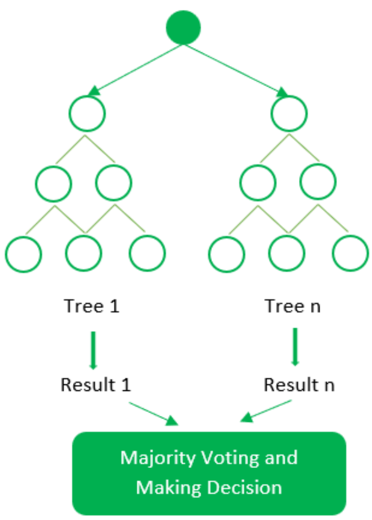

Figure 2: A schematic illustration of the random forest model [19].

As a result, none of the trees in the forest has access to the complete training dataset. Instead, the data is divided recursively into partitions, and at each node, a question is posed about an attribute to facilitate the split. The selection of the splitting criterion relies on impurity measures such as Shannon Entropy or Gini impurity.

The quality of the split in each node is evaluated using the Gini impurity function. The Gini impurity at a particular node, denoted as N, is calculated as:

$$g(N) = \sum_{i \neq j} P(w_i)P(w_j) \qquad (5)$$

where $P(w_i)$ is the proportion of the population with class label i. Shannon Entropy is another metric that can be utilized to assess the quality of a split. It quantifies the level of disorder present in the information content. In the context of decision trees, Shannon entropy is employed to determine the level of unpredictability in the information stored within a specific node of a tree. Specifically, it gauges the degree of heterogeneity of the population within that node. The entropy in a node N can be calculated as follow:

$$H(N) = -\sum_{i=1}^{i=d} P(w_i)\, log_2(P(w_i)) \qquad (6)$$

where d is the number of classes considered and $P(w_i)$ is the proportion of the population labeled as i. When all the classes are present in equal proportion in a node, entropy is at its highest. Conversely, entropy is at its lowest when there is only one class present in a node, indicating that the node is pure. The most straightforward method to choose the best splitting decision at a node is to minimize impurity as much as possible. In other words, the best split is identified by the highest information gain or the greatest reduction in impurity. The information gain due to a split can be calculated as follows:

$$\Delta I(N) = I(N) - P_L * I(N_L) - P_R * I(N_L) \qquad (7)$$

where I(N) is the impurity measure (Gini or Shannon Entropy) of node N, P_L is the proportion of the population in node Nthat goes to the left child of Nafter the split and similarly, $P_R$is the proportion of the population in node Nthat goes to the right child after the split. N_L and N_R are the left and right child of N respectively.

Bootstrap aggregating, or bagging, is a fundamental technique in ensemble machine learning algorithms that enhances the stability and accuracy of learning algorithms by reducing variance and overfitting, which are common issues when constructing decision trees. This method creates B new sets, each of size n 0, by randomly sampling from the original dataset D of size n with replacement.

3.4. Long-short Term Memory (LSTM)

3.4.1. Neural Network

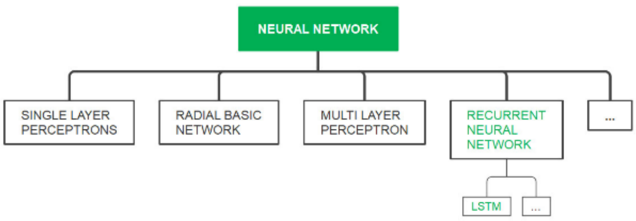

Figure 3: Overview of Neural Network.

Neural networks are a subset of machine learning and the heart of deep learning. Their name and structures are inspired by the human brain, mimicking how biological neurons signal to one another.

Neural networks comprise node layers containing input layers, one or more hidden layers, and output layers. Each node, or artificial neuron, connects to another and has an associated weight and threshold. If the output of any individual node is above the specified threshold value, that node is activated, sending data to the next layer of the network. Otherwise, no data is passed along to the next layer of the network.

3.4.2. Recurrent Neural Network (RNN)

RNN is a subtype of Neural Network [21]. RNN is suitable for processing data in sequence formatting, this idea is clearly explained as Figure 4.

In the normal neural network, we could input a layer of information at the same time. However, sometimes the information the inputs are sequential, which leads to the situation that the result would be different when we change the order.

We need a neural that could process the sequential information to fix this problem.

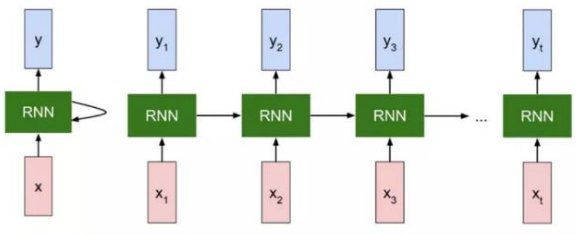

Figure 4: The internal structure of RNN.

The RNN model would be better explained in Figure 5.

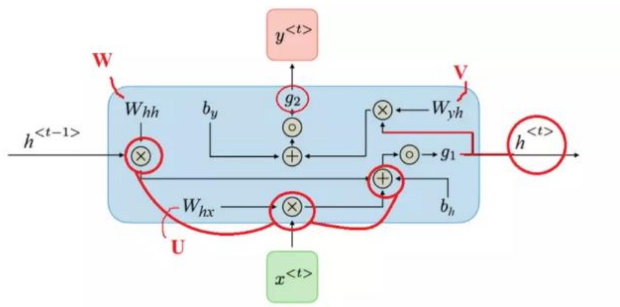

Figure 5: Basic architecture of RNN.

x representing sequential inputs (separated by time step).

x<t> representing the time step at t, and y<t> is the output of the step.

Hidden state: h<t> is the memory of the neural network. This is the combination of the previous memory (h<t-

1>) and the input at time step t x<t>.

Output for each time step y<t>: at each step (t), we have two outputs, y<t> is the output at each step, while h<t> is the summary of information, and continue to be the input of following processes.

3.4.3. The limitation of RNN Recurrent Neural Network

In this model, we can easily recognize that the output of step 1 receives the input of step 1 as well as the previous information. It has created the success of neural networks thanks to its capability of inferring. However, this capability does not work well under complex situations requiring more information for the inferring process. The limitation is context and distant dependence.

Sometimes, the models don't ultimately receive the information in the process of obtaining information. In a simple context, it can predict well. For example, “I'm a learning machine …”, it would be easy to predict the following word “learning”. However, in a complex context such as “I’m Vietnamese, I live in Ho Chi Minh City, I speak…”. The process of identifying missing words would be dependent on much more previous information. In this context, it is hard for RNN to predict well. To fix this problem, RNNs need the capability of learning in complex contexts and control the long-term dependency.

3.4.4. LSTM

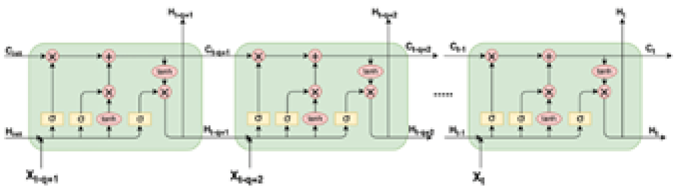

Figure 6: The internal structure of an LSTM [22].

The structure of LSTM is related to RNN. However, there is a much more complex structure, which includes 4 layers with forget gate (ft), input gate (it), output gate (ot). These gates decide which types of information would be stored, eliminated, edited, and transferred.

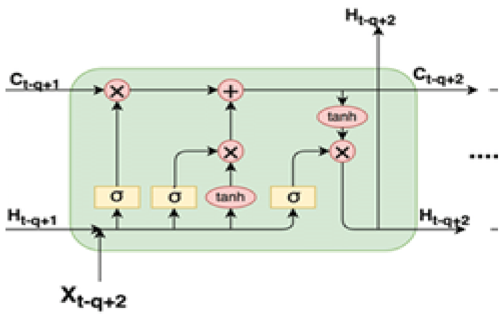

Figure 7: Basic architecture of LSTM.

Cell state (ct) and hidden state (ht) are the outputs for each module.

Cell states are impacted by the forget gate to eliminate unnecessary information and include new data from the input gate and the hidden layer of the previous module. The process of choosing information as input for the hidden layer is based on the experience of epochs. At the same time, ht is inherited from the output gate cell state. For this inheritance, LSTM is suitable for time series processing.

Table captions appear centered above the table in upper- and lower-case letters. When referring to a table in the text, no abbreviation is used and "Table" is capitalized.

4. Methodology

4.1. The Overall Research Process

The stock market is affected by many factors. To accurately predict the changes in the stock price of individual stocks, it is essential to grasp the relevant information of the stock market effectively. In this paper, we proposed a method that tries to analyze sentiments in news from 2 websites (Vietstock and Cafef) as the experience after fundamental analysis; and download the historical price of stocks as the technical analysis.

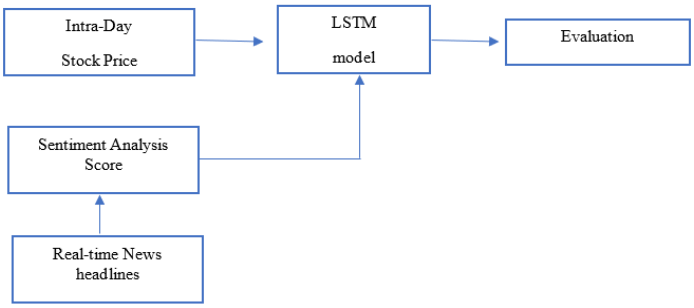

Figure 8: Research Design.

4.2. Data Collection

4.2.1. Stock closing price

For predicting stock following closing prices, we need to collect the records of ACB stock price over a while. The data source is from Vietstock, recognized as the most reliable data source. The data is collected from 1/1/2012 - 1/10/2022, with a total of 2721 observations.

4.2.2. News articles

The selected websites are Vietstock, Cafef, and Vnexpress. The reasons for choosing these sources are explained as they are described as the most reliable source in the industry and are most attractive to the investors (which could be related to the high number of visitors and references from the similar web). To collect the website links, we utilize the searching tools on these websites and include the keywords “ACB”, “lai suat ngan hang”, “ngan hang nha nuoc”, “VN index”, “VN 30” to collect related news. The reasons for choosing these keywords as we only want to include the information the is related to the targeted ACB Bank, which is in the finance and banking sector, and some crucial information about the financial market (“lai suat ngan hang”, “ngan hang nha nuoc”, “VN index”, “VN 30”). After saving the news links. We continue crawling news titles and summaries from the links. The results would be saved in text format.

4.3. Preprocessing Sentiment Analysis

News preprocessing includes five steps: lowering the characters and removing special characters; removing meaningless news; based on rule-based, and categorizing news; using PhoBert pre-trained models for sentiments analysis; aggregating information in one day into a single sentiment scoring. As screening over the news, we acknowledge that some news doesn’t contain any valuable information, which could be the news that ACB is paying to promote for themselves or even the news to announce the financial reporting without any summary. For that news, we decided to eliminate them from the pools by executing some rule-based.

4.4. Evaluation Metrics

In this paper, we decide to use the R-squared and RMSE.

Coefficient of determination (R2 or R-squared)

$$R^2 = 1 - \frac{\sum_{i=1}^{m}(X_i - Y_i)^2}{\sum_{i=1}^{m}(Y - Y_i)^2} \qquad (8)$$

Root-mean square error (RMSE)

$$RMSE = \sqrt{\frac{1}{m}\sum_{i=1}^{m}(X_i - Y_i)^2} \qquad (9)$$

5. Empirical Result

5.1. Decision Tree and Random Forest

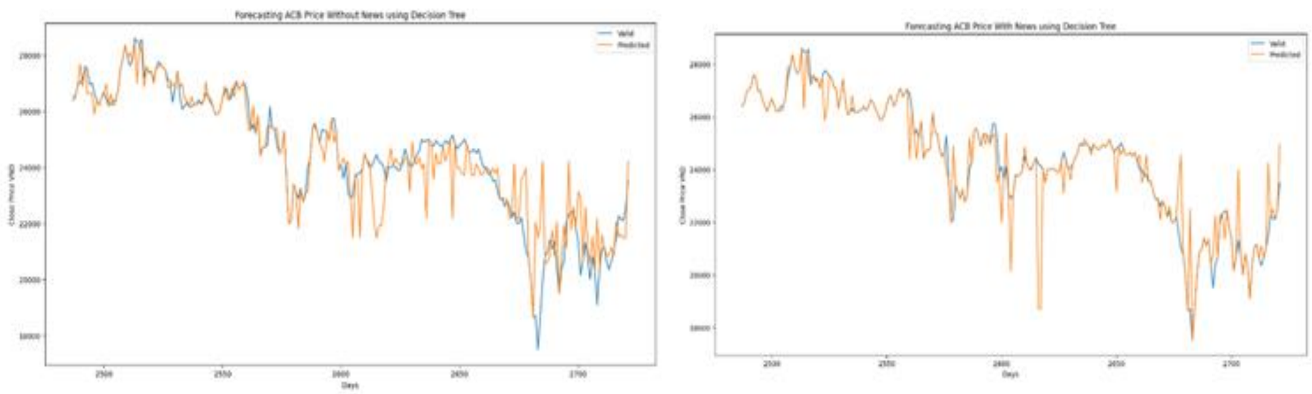

Figure 9: Decision Tree comparison between actual and predicted outcomes.

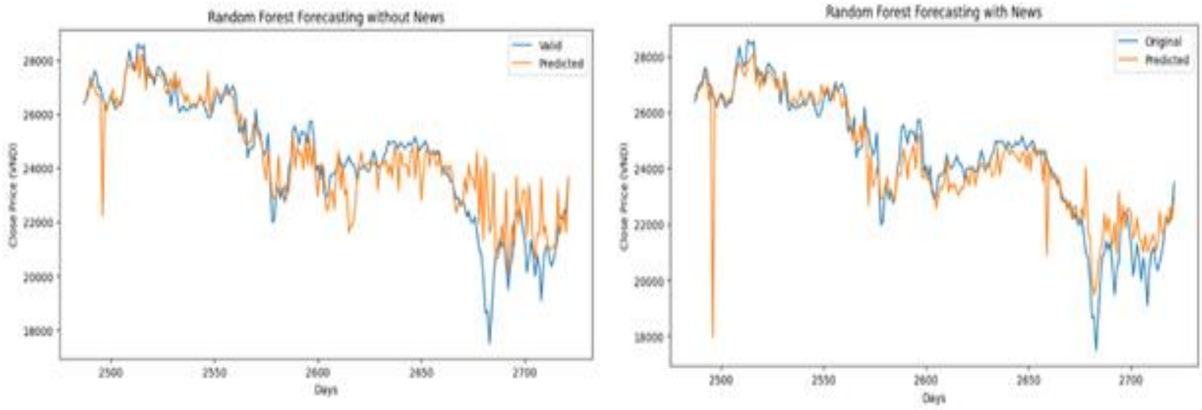

Figure 10: Random Forest comparison between actual and predicted outcomes.

5.2. LSTM

- Activation = relu

Activation function plays the role of transforming activation sum input of nodes, and state precisely the output of the node. Linear activation functions, which apply no transform, is preferred to predict a quantity.

With nonlinear activation functions, models learn more complex data structures.

The rectified linear activation function (ReL) and rectified linear activation unit (ReLu), which was emphasized in papers since 2009, improves the sensitivity to the summed activation, and reduces saturation.

The improvement is compared to tanh and sigmoid, two traditional popular nonlinear activation functions. Further, using ReLU in hidden layers can solve the vanishing gradients problem [23].

- Unit = 50, timesteps = 3, epochs = 200, batch_size = 2. After testing multiple values of these parameter, we choose the value that return the best prediction result
- Optimizer = Adam

Kingma and his colleagues (2014) presented an algorithm that combines the benefits of both the Adaptive Gradient Algorithm and Root Mean Square Propagation. The resulting algorithm not only yields high-quality results, but also achieves rapid convergence [24].

- Loss = MSE

The Mean Squared Error, or MSE, is the most popular loss function of regression model.

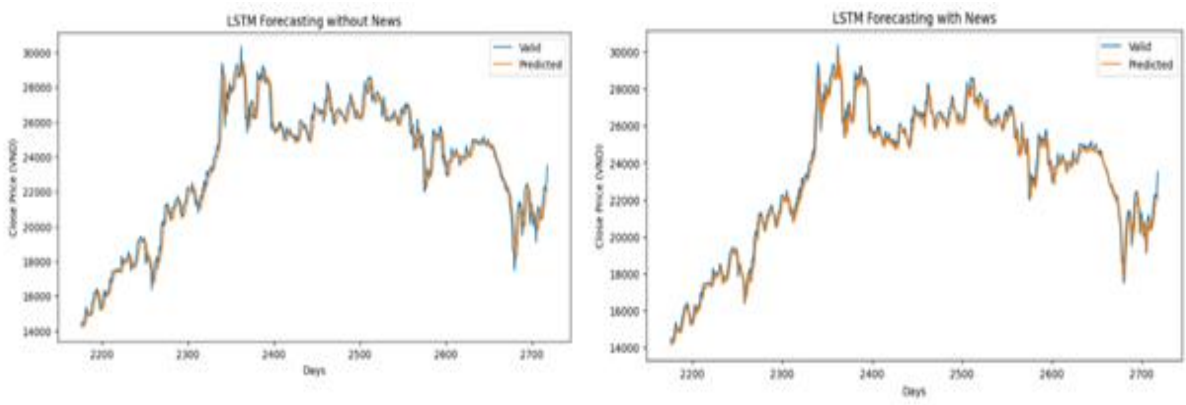

Figure 11: LSTM comparison between actual and predicted outcomes.

Base on R-squared LSTM shows the best result both in “without news” & “with news”. Models with news features have R-squared of 0.9732 (0.09 better than standard time-series models) and RMSE of 607.922 (lower than the standard model). News variable helps improved the predicting result of both Random Forest and LSTM.

Table 1: Comparison Table of Model Outcomes.

		R^2	RMSE
Decision Tree	**Without News**	0.8377	923.7732
	With News	0.8640	845.7135
Random Forest	**Without News**	0.8067	1008.148
	With News	0.8898	761.113
LSTM	**Without News**	**0.9674**	**671.130**
	With News	**0.9732**	**607.922**

6. Concluding remarks

This study examines the dynamics of the ACB stock over the period spanning 2010 to 2022 by leveraging learning-based techniques. The central aim of this research is to evaluate the feasibility of predicting ACB's price based on either its historical price information or news sentiment data or a combination of both. To accomplish this goal, we employ various machine learning techniques such as Decision Tree, Random Forest, and LSTM algorithms. The findings of this study demonstrate that the most effective approach for forecasting ACB's price is to integrate historical price data and news sentiment analysis. Specifically, the LSTM model in combination with PhoBert sentiment analysis exhibits the highest level of accuracy and predictive power, as measured by the RMSE and R2 metrics in comparison to other models.

We can suggest several research directions based on the findings of this project. First, we would like to use more machine learning models like Boosting method to forecast ACB's stock price movement. In this way, we can select the best model to predict ACB's stock price movement.

Second, the accuracy of news labeling using Phobert has not been considered. We suggest adding the percentage change labeling method to improve forecasting value.

Finally, we would like to sort expected news from unexpected news and evaluate the price movement based on each type of news. These directions will help us gain a deeper understanding of the dynamics of the financial market.

Future research can include data from various stock markets, longer time series or using a different resampling technique. Then, the LSTM network can be improved using different integrated models, hybrid models or even replaced by other updated networks to predict the stock price.

References

[1] Basu S. (1983). The relationship between earnings yield, market value and return for NYSE common stocks: Further Evidence. Journal of Financial Economics, 12(1), 129-156

[2] Jaffe J, Keim D. B. and Westerfield R. (1989). Earnings yields, market values, and stock returns. Journal of Finance, 44, 135-148.

[3] Rosenberg, B., Reid, K. and Lanstein, R. (1985). Persuasive evidence of market inefficiency. Journal of Portfolio Management, 11, 9-17.

[4] Jarrett, J. E. and Kyper, E. (2011). ARIMA modeling with intervention to forecast and analyze Chinese stock prices. International Journal of Engineering Business Management, 3(3), 53-58.

[5] Adebiyi A., Adewumi A. O. and Ayo C. K. (2014). Stock price prediction using the ARIMA model. Proceedings of the International Conference on Computer Modelling and Simulation, Cambridge, UK, pp. 105-111

[6] Hu, Z., Zhao, Y., & Khushi, M. (2021). A survey of forex and stock price prediction using deep learning. Applied System Innovation, 4(1), 9.

[7] Li, X., Wu, P., & Wang, W. (2020). Incorporating stock prices and news sentiments for stock market prediction: A case of Hong Kong. Information Processing & Management, 57(5), 102212.

[8] Dey, L., & Verma, I. (2018, January). Enterprise contextual awareness from news and social media—A framework. In 2018 10th International Conference on Communication Systems & Networks (COMSNETS) (pp. 89-96). IEEE.

[9] Ab. Rahman, A. S., Abdul-Rahman, S., & Mutalib, S. (2017). Mining textual terms for stock market prediction analysis using financial news. In Soft Computing in Data Science: Third International Conference, SCDS 2017, Yogyakarta, Indonesia, November 27–28, 2017, Proceedings 3 (pp. 293-305). Springer Singapore.

[10] aggi, M., Mandal, P., Narang, S., Naseem, U., & Khushi, M. (2021). Text mining of stocktwits data for predicting stock prices. Applied System Innovation, 4(1), 13.

[11] Bi, J. (2022). Stock Market Prediction Based on Financial News Text Mining and Investor Sentiment Recognition. Mathematical Problems in Engineering, 2022.

[12] Thanh, H. T., & Meesad, P. (2014). Stock market trend prediction based on text mining of corporate web and time series data. Journal of Advanced Computational Intelligence and Intelligent Informatics,

18(1), 22-31.

[13] Tran, T., Long, N. N., Tung, N. S., Thao, N. T., Tham, P. T. H., & Nguyen, T. Neu-stock: Stock market prediction based on financial news. In Proceedings of the 2nd International Conference on Human-centered Artificial Intelligence (Computing4Human 2021). CEUR Workshop Proceedings, Da Nang, Vietnam (Oct 2021).

[14] Le Hong, H., Nguyen, N. N., Nguyen, T. L., Nguyen, L. D., & Nguyen, N. H. (2022). Stock Market Prediction: The Application of Text-Mining in Vietnam. VNU JOURNAL OF ECONOMIC AND BUSINESS, 38(2).

[15] Tran, T., Long, N. N., Tung, N. S., Thao, N. T., Tham, P., & Nguyen, T. NEU-Stock: Stock market prediction based on financial news. In Proceedings of the 2nd International Conference on Human-centered Artificial Intelligence (Computing4Human 2021). CEUR Workshop Proceedings, Da Nang, Vietnam (Oct 2021).

[16] Nguyen, D. Q., & Nguyen, A. T. (2020). PhoBERT: Pre-trained language models for Vietnamese. arXiv preprint arXiv:2003.00744.

[17] Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), 4171-4186.

[18] Li, R., Ma, M., & Tang, N. (2023, February). Stock Price Prediction Based on Decision Trees, CNN and LSTM. In Proceedings of the 4th International Conference on Economic Management and Model Engineering, ICEMME 2022, November 18-20, 2022, Nanjing, China.

[19] Nabipour, M., Nayyeri, P., Jabani, H., Mosavi, A., Salwana, E., & S., S. (2020). Deep Learning for Stock Market Prediction. Entropy, 22(8), 840. https://doi.org/10.3390/e22080840

[20] Khaidem, L., Saha, S., & Dey, S. R. (2016). Predicting the direction of stock market prices using random forest. arXiv preprint arXiv:1605.00003.

[21] Nielsen, M. A. (2015). Neural networks and deep learning (Vol. 25, pp. 15-24). San Francisco, CA, USA: Determination press. Bai, L., Yao, L., Kanhere, S. S., Wang, X., Liu, W., & Yang, Z. (2019, November).

[22] Spatio-temporal graph convolutional and recurrent networks for citywide passenger demand prediction. In Proceedings of the 28th ACM International Conference on Information and Knowledge Management (pp. 2293-2296).

[23] Maas, A. L., Hannun, A. Y., & Ng, A. Y. (2013, June). Rectifier nonlinearities improve neural network acoustic models. In Proc. icml (Vol. 30, No. 1, p. 3).

[24] Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.