Boosting the Accuracy of Stock Market Prediction using XGBoost and Long Short-Term Memory

Agustinus Bimo Gumelar
Fakultas Ilmu Komputer
Universitas Narotama
Surabaya, Indonesia
bimogumelar@ieee.org

Haryati Setyorini
Dept of Management
STIE PERBANAS Surabaya
Surabaya, Indonesia
haryati.setyorini@perbanas.ac.id

Derry Pramono Adi
Fakultas Ilmu Komputer
Universitas Narotama
Surabaya, Indonesia
derryalbertus@ieee.org

Sengguruh Nilowardono
Economic and Business Faculty
Narotama University
Surabaya, Indonesia
sengguruh@narotama.ac.id

Latipah
Faculty of Computer Science
Narotama University
Surabaya, Indonesia
latipah.rifani@narotama.ac.id

Agung Widodo
Fakultas Ilmu Komputer
Universitas Narotama
Surabaya, Indonesia
agung.widodo@narotama.ac.id

Achmad Teguh Wibowo
Fakultas Ilmu Komputer
UIN Sunan Ampel
Surabaya, Indonesia
atw@uinsby.ac.id

MY Teguh Sulistyono
Fakultas Ilmu Komputer
Universitas Dian Nuswantoro
Semarang, Indonesia
teguh.sulistyono@dsn.dinus.ac.id

Evy Christine
Faculty of Business
Widya Mandala Catholic University
Surabaya, Indonesia
evy@ukwms.ac.id

***Abstract*— Stock exchange is one of the famous economical strategy that finally find its way to be experimented with ever-growing Machine Learning (ML) algorithm. With ML, many aspects regarding stock is learnable, to the point where one can predict stock prices. Although tempting, stock price prediction is still a challenging task due to its natural dynamic and real-time movement. Thus, predicting stock prices are deemed unseemingly. On the other hand, different patterns of stock prices are capable of represent a whole lot of detailed data, which is in favor for Deep Learning. In this study, we conducted an experiment of predicting the close stock price for 25 companies. To ensure data reliability and regional notion, these selected companies are officially enlisted in the Indonesia Stock Exchange (IDX). The two ML algorithms used for this experiment are the Long Short-Term Memory (LSTM) and Extreme Gradient Boosting (XGBoost), both known for its high accuracy of prediction from various representative data. By setting two thresholds, we were able to present a trading approach: when to buy or when to sell. This prediction result from the ML algorithm using in the ensuing trading approach leads to distinct aspects of benefit. In this experiment, XGBoost shown best performance by 99% prediction accuracy result.**

Keywords—Stock Market Prediction; Indonesia Stock Market; Meta Algorithm; XGBoost; LSTM

I. INTRODUCTION

The stock market is an ever-growing investment instrument in not-a-game-of-chances business consisted with many unpredictable moments [1]. The stock exchange has always tended to be a risky arena for those outside banking and numbers. Many are perplexed and find it difficult to comprehent its inherent yet comprehensive data. The data stream itself is ever-changing in real-time, makes it harder for traders to build a profitable strategy. This type of time series forecasting data is deemed one of the most troublesome task [2], [3]. However, a prediction strategy is doable by Machine Learning (ML), as in fact, the stock market generates more data overtime. The ability of ML algorithm to predict with high precision for many field is astonishing [4]–[6], including of providing insight for traders of when it is the right time to buy stock at low price or when to sell stock at high price and ultimately, gained profit [3], [7]–[9].

As one of the well-known and famous financial economics theories, Efficient Market Hypothesis (EMH) took information for the prices of the securities. These prices already reflect most information and according many researches [1], [9], [10], are hardly outperforms the market. EMH is known for its three variants, namely low form is the weak one, a semi-solid form is the semi strong one, and a solid form is the strongest one. The first form is the weakest one, expresses all information collected from previous trading records of shares sector. The second a semi-solid form which is semi strong one, have continuously adapting to newly available information, changing prices' values almost constantly. At last, the strong form requires insider or private information regarding each company, to make a better custom-made model suitable for traders or even other companies [10], [11]. Nevertheless, the EMH is oftentimes to be argued about due to its controversial findings [11]. An excellent example that are causing questions regarding EMH is Warren Buffet, who have earned continuous profit with an interminable period of time and really is consistent in outperforming the chaotic stock market. The beginning era of Artificial Intelligence (AI) with its hungry-of-large-data character, has pushed researchers over the years to automate the process of calculating necessary strategy to act in the stock market. An increased computational capabilities of AI shines a light in decision-making step for traders and arguably Warren Buffet in his practices, as AI is able to forecast the stock prices; it thrives on real and naturally huge number of data [3], [7], [12].

In this experiment, we used the Python as the overall programming language, as it is easily adapting to data and its

978-1-7281-9068-6/20/$31.00 ©2020 IEEE

feasibility to be implemented with available pre-built models [3], [7], [12]. Specifically, we built the Long Short-Term Memory (LSTM) model. While the LSTM being compared to traditional neural network, the LSTM has a success attributed in central component to time series forecasts, namely the memory portion. In terms of performance comparison, we also compared LSTM with the Extreme Gradient Boosting (XGBoost), which is also a versatile ML algorithm. We compared the XGBoost and LSTM algorithm to get some comparable result which is best in the times series case of forecasting.

To be compliant to real data, we provided historical stock market data of 25 different companies, which all are listed in Indonesia Stock Exchange (IDX). Both ML algorithms are fed by historical stock market data to predict the future value of stocks.

We disclosed each sections in this paper as follows: Section I explains the behavior aspect on stock market world; the chaotic, random, and its very unpredictable condition. Section II presented material and algorithms, instead of data collection and proposed technique to solve this problem. Section III show some experimental results from both LSTM and XGBoost respectively. The final conclusion and future work is provided in Section V.

II. Materials and Algorithms

In this section, we elaborates the dataset being used for the experiment, and also what it represents for. The algorithms being used is both versatile ML algorithm, namely, LSTM and XGBoost.

A. Stock Market Data from Indonesia Stock Exchange

In this experiment, we thrive to predict the closing prices of 25 companies listed in Indonesia Stock Exchange (IDX), which is adjusted in daily constant. We used the data from the previous n days. The data being fed is ranging from 2000 until 2019 and are available in the Kaggle, the platform of public dataset of various representative data [13].

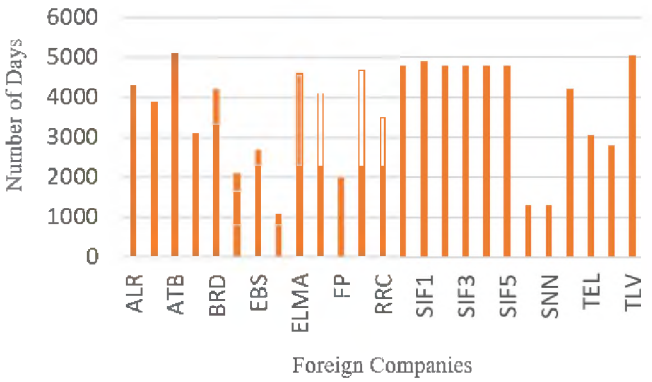


Fig. 1. Plotting stock from each company in single days

The indicators of marketing trading are using for the features, e.g.: the prices from minimum and maximum, opening and closing price, at last the price volumes [8], [13]. Fig. 1 plotted the example of foreign stock data in days duration.

B. XGBoost as the Meta Algorithm

The Extreme Gradient Boosting (XGBoost) is one of the most favorable ML techniques [14], [15]. It has numerously showing admirable performance using various of data [14], [16]. Gradient Boosting is a weak classifier (WC) convertion process into a much stronger classifier (SC). This process is repeated according to the needs of the desired model, which in this case is the stock market prediction. The "boosting" terms is inline with the engineering goal to maxed out the boosted tree algorithms performance from the computational resources, even to push its limit. Ever since its introduction in 2014 [15], and various outstanding experiments report of XGBoost, it has been one of the most sought out algorithm in implementation.

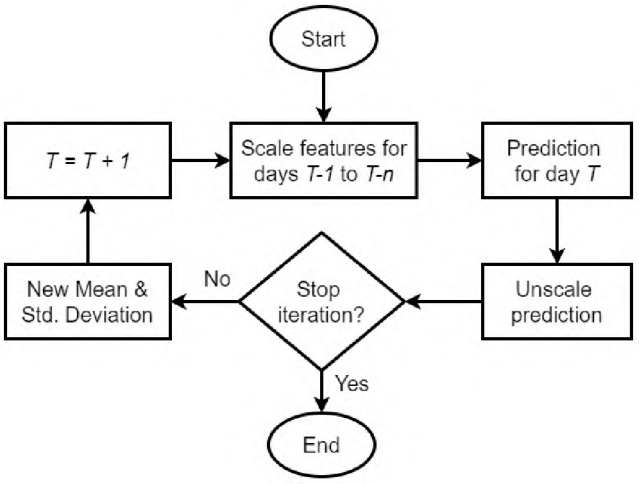


Fig. 2. Logic implementation by recursive forecasting

Tianqi Chen is the creator of the XGBoost, reporting good value of the algorithm by using gradient descent algorithm to ensemble a set of Classification And Regression Trees (CART) [16]. The CART outperforms one tree, showing powerful predictive ability. In essence, the trees acts as judges in the court, voting altogether for the most likely result. It is continuously adding new models and correcting the errors of previous models, ultimately minimizing loss [16]. A decision tree is a man-made pattern that begins from a single non-leaf node that divides into various outcomes. The outcomes would later connecting to different outcomes, in a divisive branch fashion. Every non-leaf node depicts the test on a particular feature, every branch depicts the outcome of the mentioned feature, and every each leaf node deposits a classification. If disbanding is completed for will element, the one with the least loss is deemed to be the better disbanding criterion and will be set as the node that has a rule.

$$F_{start} = 0 \quad (1)$$

$$F_{time}(x) = F_{time-1}(x) + p(x) \quad (2)$$

$$p = new\ decision \quad (3)$$

$$Objective(F_{time}) = \Omega(F_{time}) + L(F_{time-1} + F_{time}) \quad (4)$$

The disbanding cycle continues to take place on a periodic basis until the termination condition is fulfilled. In short, the set of WCs can create a single SC, as the WCs does far better than random classifiers, and the SC are the result of correcting WCs based on new input data [16].

For boosting techniques, the additive training approach is used in each phase where a week classifier is applied to the model. L is the loss feature that defines the predictive capacity, and Ω is the regularization feature that acts as an overfit controller.

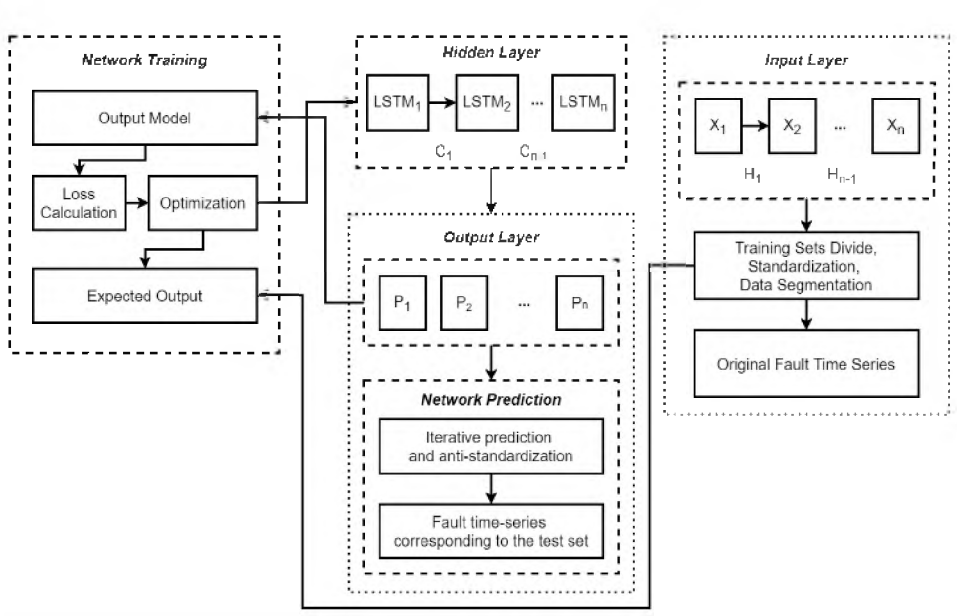


Fig. 3. General LSTM Architecture [17]

C. *LSTM Architecture for Stock Market Modeling*

In the past, the Recurrent Neural Network (RNN) has been developed into new type of neural network architecture, and it being hype to every community, it called the Long Term-Short Memory (LSTM) algorithm [18]–[20]. The LSTM algorithm is capable on handling the learning process on time series-based forecast, for a very long-term dependencies. It also work best with large type of problems. Along the way, LSTM eventually receive a well-contributed modification by modern researchers. General RNN structure was designed like a chain, reiterating back to previous layers.

When in fact, LSTM have four network layers specially interacting with each other. In general, LSTM is augmented by the "forget" gates. These gates is controlled as any error can be backpropagated through any number of layers. Such mechanism enables LSTM to learn from previous steps, reducing error effectively. Fig. 3 shown the general LSTM architecture.

$$f_t = \sigma(W_f \times [h_{t-1}, x_t] + b_f) \quad (5)$$

$$\Delta C_t = tanh(W_C \times [h_{t-1}, x_t] + b_C) \quad (6)$$

$$\mathrm{i}_t = \sigma(W_i \times [h_{t-1}, x_t] + b_i) \quad (7)$$

$$C_t = f_t \times C_{t-1} + i_t \times \Delta C_t) \quad (8)$$

$$o_t = \Sigma(W_o \times [h_{t-1}, x_t] + b_o) \quad (9)$$

$$h_t = o_t \times tanh(C_t) \quad (10)$$

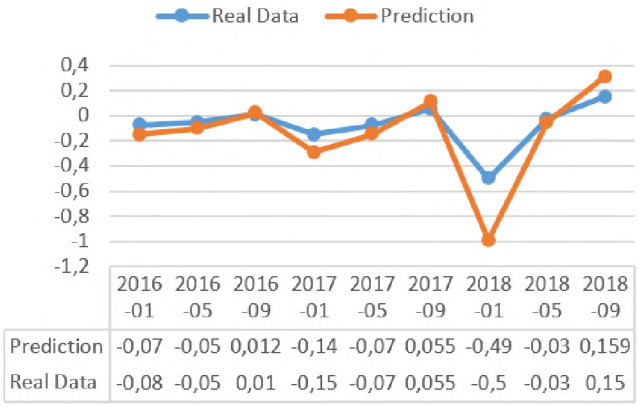
III. Experimental Results

Before we deliver the data into XGBoost and LSTM, the data were first properly break into preparation data, testing data and evaluation data. Training data or staging of data preparation is used for parameter tuning in both algorithm, whereas testing data were fed as both algorithm yielded promising result. Training data included 60% of the total instances, validation data included 20% of the total instances, and training data have the rest of instances, namely 20% of overall instances.

It is an open secret that Deep Learning (DL) poses problems as the huge architecture and its data hungry behavior, which impacted directly on computational cost and computational time. However, this was not the case for both algorithm we applied in this study.

A. *Prediction using XGBoost*

The obvious strategy of XGBoost in our experiment is as follows: the learning model of XGBoost is built upon the training data, while we tune out every hyperparameters using the validation data, and test data is fed for final result.



	2016 -01	2016 -05	2016 -09	2017 -01	2017 -05	2017 -09	2018 -01	2018 -05	2018 -09
Prediction	-0,07	-0,05	0,012	-0,14	-0,07	0,055	-0,49	-0,03	0,159
Real Data	-0,08	-0,05	0,01	-0,15	-0,07	0,055	-0,5	-0,03	0,15

Fig. 4. Plot result shows the predictions using XGBoost

Applied features are the open prices, maximum prices, minimum prices, closing prices, and the volumes. The n days applied accordingly between training data, validation data, and testing data. Fig. 4 showing the result of XGBoost stock market prediction, yielding result at 99%. The vertical axis showed the monthly Return of Investment (ROI), while the horizontal axis showed the time variable (in YYYY/MM format).

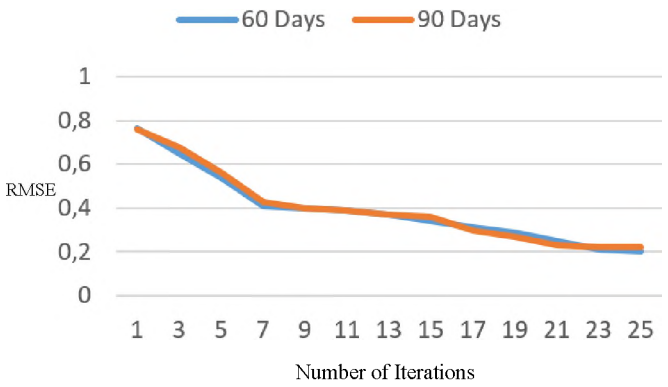


Fig. 5. Error chart in XGBoost using RMSE evaluation metric

In contrast, Fig. 5 showing error for the XGBoost algorithm. The error is in Root Mean Square Error (RMSE) metric, while presented in every iterations between duration of 60 days and 90 days.

B. Prediction using LSTM

Recently, the LSTM is a deep learning-based methodology which has been developed to deal with the issue of descent gradients in overlong continuance. According to their architecture, LSTM renew up to three gates. The first two gates, update and forget gates determine the update of each element in the memory cell [17], [21].

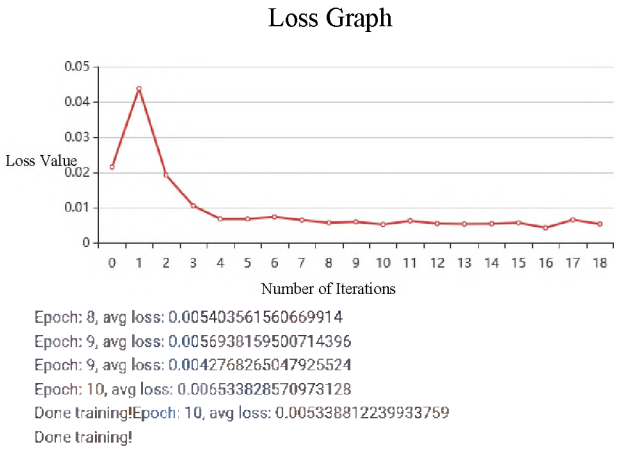


Fig. 6. Loss graph using LSTM

The next gate, namely the output gate, determines the amount of information to be shown as output for activation to the next layer. A dropout layer in between two LSTM layers are constructed to avoid overfitting.

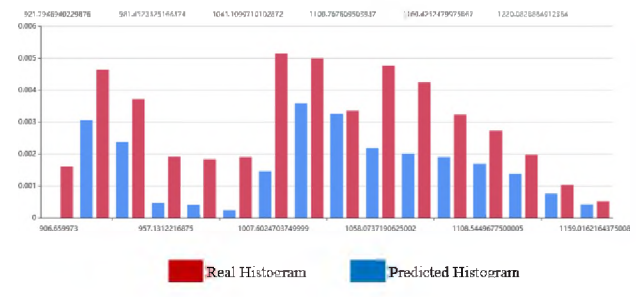


Fig. 7. Histogram chart of real data and predicted data

Fig. 6 showing loss graph of 0.005% by the last epoch. In LSTM, we used 10 epoch/learning iteration, which its improvement can be seen by the Fig. 6. In the Fig. 7, we showed a histogram chart with reality-based data, and in comparison with predicted data. We also applied a buy and sell simulation, according to our predicted histogram units.

TABLE I. BUY AND SELL SIMULATION WITH LSTM PREDICTION

Date & Month (in 2017)	Action	Price (USD)	Investment Value	Balance
7 Sept	Buy	4,679.75	0%	5,323.899535000001
14 Sept	Buy	4,625.55	-1.15%	9,949.44946
20 Sept	Buy	4,657.91	0%	5,291.549375
25 Sept	Sell	4,604.85	-1.13%	9,896.399229999999
11 Oct	Buy	4,946.25	0%	4950.149229999999
16 Oct	Sell	4,960	0.27%	9,910.149229999999
24 Oct	Buy	4,852.7	0%	5,057.449339999999
1 Nov	Sell	5,127.5	5.66%	10,184.94934
13 Nov	Buy	5,128.8	0%	5,056.199339999999

TABLE II. PERFORMANCE COMPARISON FROM PREVIOUS WORK

Author	Year	Accuracy	
		XGBoost	LSTM
Dey et al. [22]	2016	88%	-
Vargas et al. [23]	2017	-	65.08%
Roondiwala et al. [18]	2017	-	99.92%
Chatzis [16]	2018	45%	-
Cai et al. [24]	2018	-	66%
Nobre and Neves [25]	2019	49.26%	-
Basak et al. [26]	2019	94.79%	-
This proposed experiment	2020	99%	99.995%

Table I shown the buy and sell units simulation from one company, using our predicted model ranging in the year 2017. We fixed five units to be bought or sold, with a total of USD 10,000 as a fixated initial money. Table II compared the accuracy result from this work with other previous work.

IV. CONCLUSION AND FUTURE WORKS

A trading strategy is badly needed in stock market business. The stock market is never about a game of chance; it means to be an investment instrument. Thus, the decision of buying and selling units cannot be done randomly, it should be a whole profitting strategy.

Advancement of Machine Learning (ML) in the past few years have surely shine a bright light to traders. Numerous research regarding stock market, ranging from selecting best technical indicators, recognizing stock market behavior, classifying necessary strategywith selected stock market data has been done. The research that is hugely beneficial for traders are the stock market prediction. In this study, we successfully built the model of two algorithms of ML, namely the Extreme Gradient Boosting (XGBoost) and the Long Short-Term Memory (LSTM). Both have shown an outstanding accuracy in prediction result.

However, LSTM does provide a small incremental accuracy than XGBoost, by 0.005% difference. Despite having two different iterations learning number, which is 25 iterations for XGBoost and 10 iterations for LSTM, the LSTM yield optimal accuracy value in the 10th iteration. Ultimately, we have found that our model of XGBoost is able to match LSTM, which is "naturally" built for time-series data forecasting.

In the future, we set to believe in using various number of other technical features to provide much more real and detailed insight to both data and result. Furthermore, another ML algorithm with more rules and multiple features processed can be employed in order to yield a beneficial information to the traders.

REFERENCES

[1] E. Chong, C. Han, and F. C. Park, "Deep Learning Networks for Stock Market Analysis and Prediction: Methodology, Data Representations, and Case Studies," *Expert Syst. Appl.*, vol. 83, pp. 187–205, Oct. 2017.

[2] Y. S. Abu-Mostafa and A. F. Atiya, "Introduction to Financial Forecasting," *Appl. Intell.*, vol. 6, no. 3, pp. 205–213, Jul. 1996.

[3] E. L. de Faria, M. P. Albuquerque, J. L. Gonzalez, J. T. P. Cavalcante, and M. P. Albuquerque, "Predicting the Brazilian Stock Market through Neural Networks and Adaptive Exponential Smoothing Methods," *Expert Syst. Appl.*, vol. 36, no. 10, pp. 12506–12509, Dec. 2009.

[4] A. B. Gumelar *et al.*, "Human Voice Emotion Identification Using Prosodic and Spectral Feature Extraction Based on Deep Neural Networks," *IEEE 7th Int. Conf. Serious Games Appl. Heal.*, pp. 1–8, Aug. 2019.

[5] D. P. Adi, A. B. Gumelar, and R. P. Arta Meisa, "Interlanguage of Automatic Speech Recognition," in *2019 International Seminar on Application for Technology of Information and Communication (iSemantic)*, 2019, pp. 88–93.

[6] A. B. Gumelar, D. A. Lusia, A. Widodo, and R. Felani, "Using Neural Networks on Cloud Container's Performance Comparison By R on Docker (ROCKER)," *2018 Int. Symp. Adv. Intell. Informatics*, p. 5, 2018.

[7] J. Bollen, H. Mao, and X. Zeng, "Twitter Mood Predicts the Stock Market," *J. Comput. Sci.*, vol. 2, no. 1, pp. 1–8, Mar. 2011.

[8] M. Ballings, D. Van den Poel, N. Hespeels, and R. Gryp, "Evaluating Multiple Classifiers for Stock Price Direction Prediction," *Expert Syst. Appl.*, vol. 42, no. 20, pp. 7046–7056, Nov. 2015.

[9] C.-H. Cheng, T.-L. Chen, and L.-Y. Wei, "A Hybrid Model based on Rough Sets Theory and Genetic Algorithms for Stock Price Forecasting," *Inf. Sci. (Ny).*, vol. 180, no. 9, pp. 1610–1629, May 2010.

[10] A. Timmermann and C. W. J. Granger, "Efficient Market Hypothesis and Forecasting," *Int. J. Forecast.*, vol. 20, no. 1, pp. 15–27, Jan. 2004.

[11] B. G. Malkiel, "The Efficient Market Hypothesis and Its Critics," *J. Econ. Perspect.*, vol. 17, no. 1, pp. 59–82, Feb. 2003.

[12] R. Cervelló-Royo, F. Guijarro, and K. Michniuk, "Stock Market Trading Rule based on Pattern Recognition and Technical Analysis: Forecasting the DJIA Index with Intraday Data," *Expert Syst. Appl.*, vol. 42, no. 14, pp. 5963–5975, Aug. 2015.

[13] A. Wibowo, "IDX Indonesia Stock Index Price," 2019. [Online]. Available: https://www.kaggle.com/aufaawibowo/idx-indonesia-stock-price/data. [Accessed: 01-Jan-2020].

[14] P. Carmona, F. Climent, and A. Momparler, "Predicting Failure in the U.S. Banking Sector: An Extreme Gradient Boosting Approach," *Int. Rev. Econ. Financ.*, vol. 61, pp. 304–323, May 2019.

[15] R. P. Sheridan, W. M. Wang, A. Liaw, J. Ma, and E. M. Gifford, "Extreme Gradient Boosting as a Method for Quantitative Structure–Activity Relationships," *J. Chem. Inf. Model.*, vol. 56, no. 12, pp. 2353–2360, Dec. 2016.

[16] S. P. Chatzis, V. Siakoulis, A. Petropoulos, E. Stavroulakis, and N. Vlachogiannakis, "Forecasting Stock Market Crisis Events using Deep and Statistical Machine Learning Techniques," *Expert Syst. Appl.*, vol. 112, pp. 353–371, Dec. 2018.

[17] L. Lv, W. Kong, J. Qi, and J. Zhang, "An Improved Long Short-Term Memory Neural Network for Stock Forecast," *MATEC Web Conf.*, vol. 232, p. 01024, Nov. 2018.

[18] M. Roondiwala, H. Patel, and S. Varma, "Predicting Stock Prices Using LSTM," *Int. J. Sci. Res.*, vol. 6, no. 4, 2017.

[19] S. Selvin, R. Vinayakumar, E. A. Gopalakrishnan, V. K. Menon, and K. P. Soman, "Stock Price Prediction using LSTM, RNN and CNN-sliding Window Model," in *2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, 2017, pp. 1643–1647.

[20] D. Shah, W. Campbell, and F. H. Zulkernine, "A Comparative Study of LSTM and DNN for Stock Market Forecasting," in *2018 IEEE International Conference on Big Data (Big Data)*, 2018, pp. 4148–4155.

[21] A. B. Gumelar, Eko Mulyanto Yuniarno, Wiwik Anggraeni, Indar Sugiarto, A. A. Kristanto, and M. H. Purnomo, "Kombinasi Fitur Multispektrum Hilbert dan Cochleagram untuk Identifikasi Emosi Wicara," *J. Nas. Tek. Elektro dan Teknol. Inf.*, vol. 9, no. 2, pp. 180–189, May 2020.

[22] S. Dey, Y. Kumar, S. Saha, and S. Basak, "Forecasting to Classification: Predicting the direction of stock market price using Xtreme Gradient Boosting," *PESIT South Campus*, 2016.

[23] M. R. Vargas, B. S. L. P. de Lima, and A. G. Evsukoff, "Deep Learning for Stock Market Prediction from Financial News Articles," in *2017 IEEE International Conference on Computational Intelligence and Virtual Environments for Measurement Systems and Applications (CIVEMSA)*, 2017, pp. 60–65.

[24] S. Cai, X. Feng, Z. Deng, Z. Ming, and Z. Shan, "Financial News Quantization and Stock Market Forecast Research Based on CNN and LSTM," 2018, pp. 366–375.

[25] J. Nobre and R. F. Neves, "Combining Principal Component Analysis, Discrete Wavelet Transform and XGBoost to Trade in the Financial Markets," *Expert Syst. Appl.*, vol. 125, pp. 181–194, Jul. 2019.

[26] S. Basak, S. Kar, S. Saha, L. Khaidem, and S. R. Dey, "Predicting the Direction of Stock Market Prices using Tree-based Classifiers," *North Am. J. Econ. Financ.*, vol. 47, pp. 552–567, Jan. 2019.