Generating a Trading Strategy Using Candlestick Patterns with Machine Learning

Hussaina Bala Malami[1], Badamasi Imam Ya'u[2]*, Fatima Umar Zambuk[3], Mohannad Alkanan[4], Osman Elwasila[5], Mohammad Shuaib Mir[6], Mohammed Nasir Danmalam Bawa[7], Yonis Gulzar[8]*

Department of Computer Science-Faculty of Computing, Abubakar Tafawa Balewa University Bauchi, Nigeria[1, 3]
Department of Computer Science-Faculty of Science, Islamic University in Uganda, Mbale, Uganda[2]
Department of Management Information Systems-College of Business Administration, King Faisal University, Al-Ahsa 31982, Saudi Arabia[4, 5, 6, 8]
Department of Computer Science, Niger State Polytechnic Zungeru, Nigeria[7]

Abstract—**This study examines the application of machine learning (ML) algorithms for multi-day stock price prediction on the Nigerian Stock Exchange (NSE) from 2013 to 2023, to inform trading strategies. Utilizing candlestick patterns and technical indicators, including Simple Moving Average (SMA), Exponential Moving Average (EMA), and Volume Rate of Change (VROC), as input features, the models were trained to capture historical price dynamics. Among the evaluated algorithms, Ridge Regression demonstrated superior performance, achieving a Mean Absolute Error (MAE) of 0.0366 over a three-day forecasting horizon, while effectively mitigating overfitting and handling market volatility. In contrast, Decision Tree, Lasso, Support Vector Regressor (SVR), and K-Nearest Neighbors (KNN) models exhibited limitations due to sensitivity to data noise and overfitting. A recursive multi-step forecasting approach further enhanced prediction accuracy by incorporating temporal dependencies. However, backtesting revealed that predictive accuracy alone did not guarantee profitable trading outcomes, emphasizing the need to integrate market conditions, risk management, and strategy design. The findings underscore the importance of robust feature engineering and data preprocessing in financial ML applications. While Ridge Regression shows promise for stock price forecasting, successful trading strategies require a holistic framework that accounts for broader market factors. Future research should explore hybrid modeling techniques and additional exogenous variables to improve robustness.**

Keywords—Nigerian stock trading; machine learning; pattern recognition; candlestick

I. Introduction

Artificial Intelligence (AI) is revolutionizing industries by solving complex problems and boosting efficiency. It powers advanced diagnostics and personalized care in healthcare [1-2], supports intelligent decision-making and resource optimization in agriculture [3–7], and creates safer, more efficient transportation systems through autonomous vehicles and traffic control [8]. In software development, AI automates coding tasks and enhances tools, accelerating innovation. Across the board, AI is a transformative force shaping the future of global industries [9-10].

Candlestick patterns originated in 18th-century Japan, where rice traders developed them as visual tools to represent price fluctuations [11]. The foundational principles underlying these patterns, particularly their ability to encode market sentiment and signal potential reversals, were established. Later, Western scholars such as Steve Nison systematized and popularized candlestick analysis globally, bridging Eastern and Western financial traditions [12], [13], [14], [15].

Stock market prediction remains a formidable challenge due to market noise and the semi-strong form of market efficiency, which posits that asset prices reflect all publicly available information [16]. Despite skepticism from financial economists about the feasibility of consistent profits through prediction, technical analysis, especially candlestick charting, has endured as a widely used methodology [17], [18]. Candlestick charts synthesize open, high, low, and close (OHLC) prices into intuitive visualizations, capturing both supply-demand dynamics [19] and investor psychology [20], [21]. Empirical studies have demonstrated their efficacy, for instance, [15] showed that candlestick-based strategies generated statistically significant profits in S&P 500 stocks, while [22] confirmed the profitability of one-day reversal patterns [23]. Subsequent work by [24] and [25] further validated the predictive power of two-day candlestick formations, and [26], [27] extended this to three-day patterns under varying market trends [28].

However, the literature is not unanimous. Contradictory findings, such as those of [29], who found no evidence of predictive ability in DAX index futures, underscore the need for further investigation [30]. These discrepancies motivate the integration of modern computational techniques, particularly machine learning (ML), to enhance candlestick analysis [31], [32]. ML's capacity to process high-dimensional data and detect non-linear patterns makes it well-suited to refine traditional methods.

II. Related Works

The research community has already investigated combining machine learning and pattern recognition to solve next-day stock price prediction in order to "take the best from the two worlds". For instance, Kamo and Dagli suggested a fuzzy logic-based gating network that accepts information regarding candlestick patterns as input [33]. To train feed-forward neural networks and support vector machines, respectively, approaches that extract features from candlestick charts were introduced [34][16]. An

*Corresponding Author, emails: biyau@iuiu.ac.ug, ygulzar@kfu.edu.sa

auto-regressive time series forecasting model with ordered fuzzy candlestick patterns was integrated [35].

On the other hand, researchers proposed PRML, a novel candlestick pattern recognition model using machine learning methods to improve stock trading decisions [11]. Four popular machine learning methods and 11 different feature types are applied to all possible combinations of daily patterns to start the pattern recognition schedule. Different time windows from one to ten days are used to detect the prediction effect at different periods. An investment strategy is constructed according to the identified candlestick patterns and a suitable time window.

A method that decouples the machine learning and pattern recognition steps within the trading system was proposed [31]. The study leverages various machine learning models, including both shallow and deep supervised models, as well as autoregressive techniques. These models are used to generate trading recommendations based on historical stock price data and relevant features.

Two authors proposed a technique to generate stock trading strategies employing features that are shown to effectively recognize patterns in candlestick charts [36]. The features are combined into a tree-like trading strategy using the Chi-square Automatic Interaction Detector algorithm. The technique is evaluated using actual stocks from the Stock Exchange of Thailand. The results show that the generated strategies are more profitable than other popular trading techniques, such as moving average convergence divergence, exponential moving average, relative strength index, stochastic oscillator, and average directional index.

Another researcher examined whether candlesticks patterns can predict trend swings and found that well-known 2-day “Engulfing” pattern has failed to produce a positive gain, while the “Harami” pattern has barely succeeded in doing so [37]. A more complex pattern known as the “Kicker” barely achieved a positive average gain and was also outperformed by the simple Buy and Hold (B&H) strategy. It was also reported that the “Stairs” pattern proposed achieved a positive gain for all twenty examined stocks and outperformed the B&H strategy for sixteen out of the twenty stocks.

A forecasting model based on chaotic mapping, firefly algorithm, and support vector regression to predict stock market price was proposed [16]. Compared with genetic algorithm-based SVR, chaotic genetic algorithm-based SVR, artificial neural networks, and adaptive neuro-fuzzy inference systems, the proposed model performs best on mean squared error and mean absolute percent error.

In another study, a stock price predicting system by combining SVR and an ensemble adaptive neuro fuzzy inference system (ENANFIS) was proposed [38]. The experimental results showed that the SVR-ENANFIS model has superior prediction performance than ENANFIS, SVR-Linear, SVR-SVR, and SVR-ANN.

III. Materials and Methods

A. Dataset Overview

Historical OHLC (Open, High, Low, Close) data from the Nigerian Stock Exchange (2013–2023) was collected. Using TA-Lib and pandas_ta libraries, 43 candlestick patterns were extracted. Technical indicators, SMA, EMA, and VROC, were also computed. Data preprocessing included handling missing values, linear interpolation, date indexing, and feature standardization.

A total of 51 features were created from the raw data. This included both traditional indicators and patterns (e.g. Doji, Hanging Man, Hammer). These features were selected to capture price direction, market momentum, and reversal likelihood.

Five ML algorithms were evaluated, which include Ridge Regression (with regularization), Decision Tree, Lasso Regression, Support Vector Regression (SVR), and K-Nearest Neighbors (KNN). The models were trained using Skforecast and Scikit-learn, optimized with time series cross-validation. Recursive multi-step forecasting was used for predicting three-day future prices. Our performance was assessed using Root Mean Square Error (RMSE), Mean Absolute Error (MAE), and Coefficient of Determination (R^2). Furthermore, the hyperparameters for Ridge (alpha) and Decision Tree (depth, splits, leaves) were tuned using grid search.

B. Evaluation Metrics

To assess the accuracy and reliability of the machine learning models, three standard regression evaluation metrics were employed: Root Mean Square Error (RMSE), Mean Absolute Error (MAE), and the Coefficient of Determination (R^2). Each provides unique insights into the model's predictive performance.

1) Root Mean Square Error (RMSE): RMSE measures the average magnitude of the errors between predicted and actual values, giving more weight to large errors due to the squaring of differences. A lower RMSE indicates better predictive accuracy. Because it penalizes large errors more heavily, RMSE is particularly useful when large deviations are especially undesirable in applications like stock price forecasting. RMSE is given by the following formula:

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

a) Mean Absolute Error (MAE): MAE represents the average absolute difference between the predicted and actual values, treating all errors equally regardless of direction. MAE is straightforward to interpret. A lower MAE reflects higher accuracy, making it useful for understanding the average error size in the same units as the target variable (e.g., price in Naira). MAE is given by the following formula:

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}[y_i - \hat{y}_i] \tag{2}$$

b) Coefficient of determination (R^2): R^2 measures how well the model explains the variance in the actual data. It ranges from 0 to 1, with higher values indicating a better fit. An R^2 value close to 1 indicates that the model explains most of the variability in the target variable. In stock forecasting, a high R^2 suggests that the model captures significant trends and fluctuations in price data. R^2 is given by the following formula:

$$R^2 = 1 - \frac{\sum(y_i - \hat{y}_i)^2}{(y_i - \bar{y}_i)^2} \tag{3}$$

C. Proposed Model

The methodology begins with the collection of historical data from the Nigerian Stock Exchange, covering a 10-year period from 2013 to 2023. The dataset includes standard OHLC (Open, High, Low, Close) and volume data, which serve as the foundation for all subsequent analysis. Using libraries such as TA-Lib and pandas ta, various technical indicators and candlestick patterns were extracted to enrich the dataset. In the feature engineering stage, 43 candlestick patterns (e.g. Doji, Hammer, Hanging Man) were identified, alongside key technical indicators like the Simple Moving Average (SMA), Exponential Moving Average (EMA), and Volume Rate of Change (VROC). Together, these contributed to a total of 51 features, carefully designed to capture short-term price movements, trend reversals, and market momentum.

Next, the data underwent preprocessing to ensure accuracy and compatibility with machine learning models. This included handling missing values through imputation, applying linear interpolation, setting proper date indexing, and standardizing the features to bring all values into a similar range. These steps were essential to improve model learning and prevent bias caused by scale differences. The model training and selection phase involved testing five machine learning algorithms: Ridge Regression, Lasso Regression, Decision Tree, Support Vector Regression (SVR), and K-Nearest Neighbors (KNN). The training process utilized Scikit-learn and Skforecast libraries, with model hyperparameters optimized through grid search, particularly focusing on tuning aspects like Ridge's alpha and Decision Tree's depth and split criteria.

Fig. 1. The proposed model.

After training, the models were used in a multi-step recursive forecasting strategy to predict stock prices three days ahead. This approach allowed the system to simulate real trading conditions by using previous predictions as inputs for subsequent forecasts, which is particularly useful in time-series scenarios.

Finally, the models were subjected to evaluation using multiple performance metrics, including Root Mean Square Error (RMSE), Mean Absolute Error (MAE), and the Coefficient of Determination (R^2). These metrics enabled a comprehensive assessment of each model's predictive accuracy, generalization ability, and error magnitude, supporting the selection of the most reliable model for further trading strategy development. Fig. 1 describes the processes involved in the proposed model.

IV. Results and Discussion

A. Predictive Performance

Among the five machine learning models evaluated, Ridge Regression demonstrated the most consistent and accurate performance in forecasting multi-day stock price movements. With a Mean Absolute Error (MAE) of 0.0366 and a Root Mean Square Error (RMSE) of 0.0367, Ridge Regression successfully balanced bias and variance through regularization, making it well-suited for the high-dimensional, noisy data typical of stock markets. Its strong generalization capabilities make it a robust candidate for short-term forecasting in volatile environments like the Nigerian Stock Exchange (NSE).

In contrast, the Decision Tree model, despite showing near-perfect accuracy on the training dataset (very low training error), suffered from severe overfitting. The test phase MAE rose sharply to 19.43, indicating a lack of generalization and an inability to handle new, unseen data. This result highlights the Decision Tree's sensitivity to noise and its tendency to overfit in the absence of pruning or ensemble techniques.

Other models, including Lasso Regression, Support Vector Regression (SVR), and K-Nearest Neighbors (KNN), delivered moderate to poor results. These models showed higher error rates in both training and testing phases, possibly due to their sensitivity to market volatility and the relatively sparse or nonlinear nature of the input data. KNN, in particular, may have been impacted by local noise, while SVR's performance is often highly dependent on.

B. Model Training

Fig. 2 illustrates the model's performance using the Root Mean Square Error (RMSE). The Ridge model performed best, with an RMSE of 1.35, followed by the Decision Tree model, which had an RMSE of 8.71. The Lasso, SVR, and KNN models performed poorly, with RMSE values of 24.94, 62.91, and 55.04, respectively.

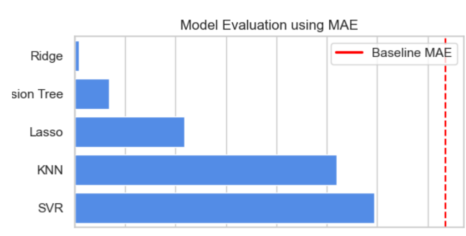

Fig. 2. Model evaluation using RMSE.

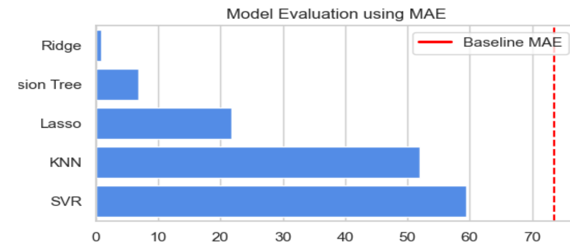

Fig. 3. Model evaluation using MAE.

Fig. 3 illustrates the model's performance using the Root Mean Square Error (RMSE). The Ridge model performed best, with an RMSE of 1.35, followed by the Decision Tree model, which had an RMSE of 8.71. The Lasso, SVR, and KNN models performed poorly, with RMSE values of 24.94, 62.91, and 55.04, respectively.

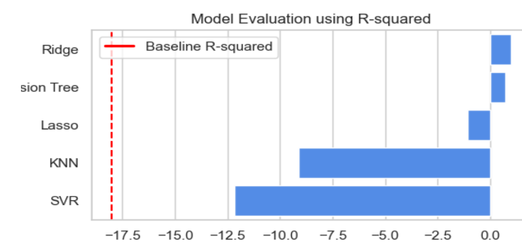

Fig. 4. Model evaluation using R-squared.

Fig. 4 shows how well the models performed using the coefficient of determination (R^2) metric. The Ridge model performed best with an R^2 of 0.99. The Decision Tree model also performed well with an R^2 of 0.75. However, the Lasso, SVR, and KNN models performed poorly, with R2 values of -1.07, -12.18, and -9.09, respectively, indicating no predictive power.

C. Trading System Implementation

Following the development of predictive models, a basic trading strategy was implemented to test the real-world applicability of the forecasts. The strategy used a fixed stop-loss of 3% and allocated 50% of available capital per trade, with signals generated from the Ridge model's predicted direction. The signal generator incorporated trend validation and position sizing logic based on forecasted returns.

However, early backtesting results indicated zero or negligible profit, despite strong predictive metrics. This disconnect between forecast accuracy and trading profitability highlights a critical gap in model-to-strategy translation. The lack of gains may be attributed to several factors: static risk parameters, market slippage, absence of transaction cost modeling, or timing mismatches between prediction and execution. These results underscore the importance of integrating dynamic risk management, execution optimization, and broader contextual signals (such as macroeconomic news or investor sentiment) into the trading framework.

D. Evaluation and Comparative Performance of Models

The comparative performance in Table I highlights how different machine learning models fared in predicting multi-day stock prices. Ridge Regression emerged as the most reliable model, achieving the lowest test MAE (0.0366) and RMSE

(0.0367). Its regularization technique helped manage overfitting, making it well-suited for the noisy, multicollinear features typical in financial data. In contrast, the Decision Tree model, while showing excellent results during training, completely failed to generalize, with its test MAE skyrocketing to 19.43, indicating severe overfitting. The overfitting occurred because the decision tree was overly complex, memorizing noise and specific patterns in the training data rather than learning generalizable trends. In financial forecasting, this limits the model's usefulness, making regularization and robust validation essential to improve performance on unseen data.

Models like Lasso Regression, Support Vector Regression (SVR), and K-Nearest Neighbors (KNN) performed moderately. Lasso offered some interpretability and handled sparse features, but at a slight cost to accuracy. SVR showed reasonable balance, though sensitive to kernel tuning, while KNN struggled with market volatility due to its reliance on proximity in high-dimensional space. Overall, the results reinforce Ridge Regression as the most stable and generalizable model for short-term stock prediction in this context. However, they also underscore the importance of avoiding overly complex or overly simple models that may not translate well into real trading scenarios.

The poor performance of Lasso, SVR, and KNN likely stems from their sensitivity to the dataset's structural characteristics. Lasso's aggressive feature selection may have eliminated key predictors in a volatile, high-noise financial dataset, leading to severe underfitting. SVR and KNN, which rely heavily on local patterns and data density, struggle when market fluctuations create irregular, non-repetitive trends, making it difficult for them to capture meaningful predictive relationships.

TABLE I THE COMPARATIVE PERFORMANCE

Model	Training MAE	Testing MAE	Training RMSE	Testing RMSE	R² Score	Notes
Ridge Regression	0.0333	0.0366	0.0341	0.0367	High	Best overall performance; balanced generalization and low variance
Lasso Regression	0.0402	0.0601	0.0417	0.0632	Moderate	Penalizes overfitting but loses accuracy with sparse signals
K-Nearest Neighbors	0.0515	0.0560	0.0527	0.0583	Low	Struggles with volatility; sensitive to data density
SVR (Linear Kernel)	0.0498	0.0534	0.0505	0.0550	Moderate	Good for stable trends, less effective under erratic price movements
Decision Tree	**0.0005**	**19.43**	0.0006	19.88	Very Low	Severe overfitting; excellent on train data, but fails to generalize

V. CONCLUSION

This study investigated the application of machine learning algorithms in forecasting stock prices on the Nigerian Stock Exchange (NSE) using historical data from 2013 to 2023. The core objective was to develop a multi-day trading strategy by incorporating both candlestick patterns and technical indicators into predictive models. The integration of indicators such as the Simple Moving Average (SMA), Exponential Moving Average (EMA), and Volume Rate of Change (VROC), alongside 43 candlestick patterns, enriched the feature space and improved the models' ability to learn meaningful trends and price movements.

Among the five machine learning models evaluated, Ridge Regression consistently outperformed others, achieving a minimum Mean Absolute Error (MAE) of 0.0366 over a three-day forecast horizon. Its strong generalization capability and robustness to multicollinearity made it particularly well-suited for volatile financial datasets. In contrast, the Decision Tree model showed signs of overfitting, while Lasso Regression, Support Vector Regression (SVR), and K-Nearest Neighbors (KNN) were less effective due to their sensitivity to market fluctuations.

The study further emphasizes that accurate predictions alone do not guarantee profitable trading outcomes. While multi-step recursive forecasting proved effective in enhancing predictive precision, the transition from forecasts to profitable trades remains complex. Factors such as real-time market conditions, execution delays, and inadequate risk management can erode trading performance. This highlights the need for integrated strategies that combine robust predictive models with sound risk management, position sizing, and decision filters.

Ultimately, the research affirms the feasibility of leveraging machine learning, candlestick analysis, and technical indicators for short-term price prediction in emerging markets. It also underscores the importance of comprehensive data preprocessing, thoughtful feature engineering, and iterative model tuning. The findings contribute to the expanding literature on AI-driven financial analytics and offer a strong foundation for future exploration into adaptive trading systems.

A. Future Work

To further improve the effectiveness and practical relevance of the study, the following areas are recommended for future research and development:

1) Enhance Feature Engineering by adding more technical indicators (e.g. MACD, RSI) and refining candlestick representations to better capture market behavior.

2) Broaden the Dataset to include various sectors or global markets, improving the model's adaptability and generalizability.

3) Adopt Deep Learning Techniques such as LSTM or TCNs to better capture time-series patterns and enhance forecast accuracy.

4) Implement Dynamic Model Updating to ensure the system adapts to evolving market conditions and avoids performance degradation.

5) Expand Indicator Selection to identify and utilize the most predictive technical signals.

6) Optimize and Combine Models using ensemble or hybrid methods to improve robustness and predictive power.

7) Integrate Risk Management features like dynamic stop-loss, position sizing, and sentiment analysis to minimize trading risk.

8) Establish Continuous Evaluation through regular backtesting, real-time validation, and model recalibration to maintain performance over time.

ACKNOWLEDGMENT

This work was supported by the Deanship of Scientific Research, the Vice Presidency for Graduate Studies and Scientific Research, King Faisal University, Saudi Arabia under the Project KFU253264.

REFERENCES

[1] N. M. Wormi, B. I. Ya'u, S. Boukari, M. A. Musa, F. Shittu, and M. A. Lawal, "Deeper Architecture for Brain Age Prediction Based on MRI Images Using Transfer Learning Technique," *Procedia Comput Sci*, vol. 212, pp. 441–453, 2022.

[2] M. A. Wakili, H. A. Shehu, M. Abdollahi, Y. B. Imam, M. H. Sharif, and H. Kusetogullari, "DenseNet-FPA: Integrating DenseNet and Flower Pollination Algorithm for Breast Cancer Histopathology Image Classification," *IEEE Access*, 2025.

[3] Gulzar, Y. PapNet: An AI-Driven Approach for Early Detection and Classification of Papaya Leaf Diseases. *Applied Fruit Science 2025 67:4* 2025, *67*, 1–11, doi:10.1007/S10341-025-01466-9.

[4] Gulzar, Y. Papaya Leaf Disease Classification Using Pre-Trained Deep Learning Models: A Comparative Study. *Applied Fruit Science* 2025, *67*, 1–10, doi:10.1007/S10341-025-01533-1/METRICS.

[5] Gulzar, Y.; Ünal, Z. Time-Sensitive Bruise Detection in Plums Using PlmNet with Transfer Learning. *Procedia Comput Sci* 2025, *257*, 127–132, doi:10.1016/J.PROCS.2025.03.019.

[6] Gulzar, Y.; Ünal, Z. Optimizing Pear Leaf Disease Detection Through PL-DenseNet. *Applied Fruit Science* 2025, *67*, 1–13, doi:10.1007/s10341-025-01265-2.

[7] Seelwal, P.; Dhiman, P.; Gulzar, Y.; Kaur, A.; Wadhwa, S.; Onn, C.W.; Goyal, N.; Shanmugasundaram, H. A Systematic Review of Deep Learning Applications for Rice Disease Diagnosis: Current Trends and Future Directions. *Front Comput Sci* 2024, *6*, 1452961, doi:10.3389/FCOMP.2024.1452961.

[8] Idris, I.Y.; Ya'u, B.I.; Ali, U.; Gulzar, Y. An Improved Deep Learning Framework Based on Multi-Scale Convolutional Architecture for Road Crack Detection. *Lecture Notes in Networks and Systems* 2025, *1165*, 109–121, doi:10.1007/978-981-97-8336-6_9.

[9] A. A. Aihong, B. I. Ya'U, U. Ali, A. Ahmad, and M. A. Lawal, "An optimized deep learning method for software defect prediction using Whale Optimization Algorithm," *Научно-технический вестник информационных технологий, механики и оптики*, vol. 24, no. 2, pp. 222–229, 2024.

[10] Z. Bala *et al.*, "Transfer learning approach for malware images classification on android devices using deep convolutional neural network," *Procedia Comput Sci*, vol. 212, pp. 429–440, 2022.

[11] Y. Lin, S. Liu, H. Yang, H. Wu, and B. Jiang, "Improving stock trading decisions based on pattern recognition using machine learning technology," *PLoS One*, vol. 16, no. 8, p. e0255558, 2021.

[12] Y. Hu, B. Feng, X. Zhang, E. W. T. Ngai, and M. Liu, "Stock trading rule discovery with an evolutionary trend following model," *Expert Syst Appl*, vol. 42, no. 1, pp. 212–222, 2015.

[13] W. Hu, Y.-W. Si, S. Fong, and R. Y. K. Lau, "A formal approach to candlestick pattern classification in financial time series," *Appl Soft Comput*, vol. 84, p. 105700, 2019.

[14] H. Hu, L. Tang, S. Zhang, and H. Wang, "Predicting the direction of stock markets using optimized neural networks with Google Trends," *Neurocomputing*, vol. 285, pp. 188–195, 2018.

[15] G. Caginalp and H. Laurent, "The predictive power of price patterns," *Appl Math Finance*, vol. 5, no. 3–4, pp. 181–205, 1998.

[16] E. Ahmadi, M. Jasemi, L. Monplaisir, M. A. Nabavi, A. Mahmoodi, and P. A. Jam, "New efficient hybrid candlestick technical analysis model for stock market timing on the basis of the Support Vector Machine and Heuristic Algorithms of Imperialist Competition and Genetic," *Expert Syst Appl*, vol. 94, pp. 21–31, 2018.

[17] R. C. Brasileiro, V. L. F. Souza, B. J. T. Fernandes, and A. L. I. Oliveira, "Automatic method for stock trading combining technical analysis and the artificial bee colony algorithm," in *2013 Ieee congress on evolutionary Computation*, IEEE, 2013, pp. 1810–1817.

[18] O. Bustos and A. Pomares-Quimbaya, "Stock market movement forecast: A systematic review," *Expert Syst Appl*, vol. 156, p. 113464, 2020.

[19] H. Bessembinder and K. Chan, "Market efficiency and the returns to technical analysis," *Financ Manage*, pp. 5–17, 1998.

[20] B. R. Marshall, M. R. Young, and L. C. Rose, "Candlestick technical trading strategies: Can they create value for investors?," *J Bank Financ*, vol. 30, no. 8, pp. 2303–2323, 2006.

[21] R. Naranjo, J. Arroyo, and M. Santos, "Fuzzy modeling of stock trading with fuzzy candlesticks," *Expert Syst Appl*, vol. 93, pp. 15–27, 2018.

[22] Y.-J. Goo, D.-H. Chen, and Y.-W. Chang, "The application of Japanese candlestick trading strategies in Taiwan," *Investment Management and Financial Innovations*, no. 4, Iss. 4, pp. 49–79, 2007.

[23] Q. Wang, W. Xu, and H. Zheng, "Combining the wisdom of crowds and technical analysis for financial market prediction using deep random subspace ensembles," *Neurocomputing*, vol. 299, pp. 51–61, 2018.

[24] M. Zhu, S. Atri, and E. Yegen, "Are candlestick trading strategies effective in certain stocks with distinct features?," *Pacific-Basin Finance Journal*, vol. 37, pp. 116–127, 2016.

[25] S. Chen, S. Bao, and Y. Zhou, "The predictive power of Japanese candlestick charting in Chinese stock market," *Physica A: Statistical Mechanics and its Applications*, vol. 457, pp. 148–165, 2016.

[26] T.-H. Lu, "The profitability of candlestick charting in the Taiwan stock market," *Pacific-Basin Finance Journal*, vol. 26, pp. 65–78, 2014.

[27] T.-H. Lu, Y.-C. Chen, and Y.-C. Hsu, "Trend definition or holding strategy: What determines the profitability of candlestick charting?," *J Bank Financ*, vol. 61, pp. 172–183, 2015.

[28] A. M. Ozbayoglu, M. U. Gudelek, and O. B. Sezer, "Deep learning for financial applications: A survey," *Appl Soft Comput*, vol. 93, p. 106384, 2020.

[29] J. H. Fock, C. Klein, and B. Zwergel, "Performance of candlestick analysis on intraday futures data," *Journal of Derivatives*, vol. 13, no. 1, p. 28, 2005.

[30] F. Zhou, Q. Zhang, D. Sornette, and L. Jiang, "Cascading logistic regression onto gradient boosted decision trees for forecasting and trading stock indices," *Appl Soft Comput*, vol. 84, p. 105747, 2019.

[31] L. Cagliero, J. Fior, and P. Garza, "Shortlisting machine learning-based stock trading recommendations using candlestick pattern recognition," *Expert Syst Appl*, vol. 216, p. 119493, 2023.

[32] L. Cagliero, P. Garza, G. Attanasio, and E. Baralis, "Training ensembles of faceted classification models for quantitative stock trading," *Computing*, vol. 102, no. 5, pp. 1213–1225, 2020.

[33] T. Kamo and C. Dagli, "Hybrid approach to the Japanese candlestick method for financial forecasting," *Expert Syst Appl*, vol. 36, no. 3, pp. 5023–5030, 2009.

[34] M. Jasemi, A. M. Kimiagari, and A. Memariani, "A modern neural network model to do stock market timing on the basis of the ancient investment technique of Japanese Candlestick," *Expert Syst Appl*, vol. 38, no. 4, pp. 3884–3890, 2011.

[35] A. Marszałek and T. Burczyński, “Modeling and forecasting financial time series with ordered fuzzy candlesticks,” *Inf Sci (N Y)*, vol. 273, pp. 144–155, 2014.

[36] S. Thammakesorn and O. Sornil, “Generating trading strategies based on candlestick chart pattern characteristics,” in *Journal of Physics: Conference Series*, IOP Publishing, 2019, p. 12008.

[37] G. Cohen, “Best candlesticks pattern to trade stocks,” *International Journal of Economics and Financial Issues*, vol. 10, no. 2, p. 256, 2020.

[38] J. Zhang, L. Li, and W. Chen, “Predicting stock price using two-stage machine learning techniques,” *Comput Econ*, vol. 57, no. 4, pp. 1237–1261, 2021.