# Comparative Analysis of Conventional AI Approaches for Stock Price Prediction: A Case Study on BAC

Oğuz DENİZ
*Artificial Intelligence Engineering*
*TOBB ETU*
Ankara, Türkiye
o.deniz@etu.edu.tr

***Abstract*—This work presents a comprehensive study comparing conventional Artificial Intelligence (AI) methods with state-of-the-art Machine Learning (ML) models for stock price prediction. Utilizing time series data of Bank of America (BAC) from 2005 to 2022 sourced from Yahoo Finance, the research explores a variety of approaches. Conventional approaches include statistical-stochastic techniques—Hidden Markov Models, Monte Carlo Simulation, and GARCH—as well as metaheuristic algorithms like Genetic Algorithm and Colony Optimization. An agentic approach is also explored through 13 tradebot agents, each with a unique buy-sell strategy. The performance of these methods is benchmarked against established ML models, such as ARIMA and Prophet, to assess forecasting accuracy and the ability to capture complex market dynamics, thereby revealing trade-offs between computational complexity and prediction precision.**

***Index Terms*—Artificial Intelligence, Stock Price Prediction, Algorithmic Trading**

## I. Introduction

Stock market prediction remains a significant research area due to its economic impact and the inherent complexity of financial time series, which are characterized by non-linearity, volatility, and noise. This study examines conventional Artificial Intelligence (AI) methods for stock price prediction and compares them with state-of-the-art Machine Learning (ML) models. Historical data for Bank of America Corporation (BAC) from 2005 to 2022, sourced from Yahoo Finance, serves as the basis for analysis.

Conventional techniques include statistical-stochastic methods—such as Hidden Markov Models, Monte Carlo Simulation, and GARCH—and metaheuristic algorithms including Genetic Algorithm and Colony Optimization. Additionally, 13 tradebot agents, each employing a unique buy-sell strategy, are introduced as an agentic approach to simulate algorithmic trading. Their performance is benchmarked against established ML models like ARIMA and Prophet, with a focus on prediction accuracy, computational complexity, and adaptability to market dynamics. This study contributes to a deeper understanding of the practical applications and limitations of both conventional AI and ML techniques in financial market prediction, thereby offering valuable insights for researchers and practitioners in the field.

## II. Related Works

The literature on stock price prediction has explored a variety of Artificial Intelligence (AI) and Machine Learning (ML) techniques to address the challenges of forecasting financial markets. Thio-ac et al. [1] demonstrated the use of Hidden Markov Models (HMMs) combined with quadratic programming for stock price prediction and portfolio optimization, emphasizing the ability of HMMs to capture market regimes. Similarly, Mutinda and Amos [2] employed combined GARCH-AI models to forecast volatility and manage risk, highlighting the integration of econometric models with AI.

A bibliometric study [3] has mapped the evolution of AI applications in stock prediction, offering insights into the trends and research directions in this field. Complementing this, Sezer et al. [4] conducted a systematic literature review on deep learning models for financial time series forecasting, categorizing studies by application areas and DL model choices, and identifying both setbacks and opportunities in the field.

The application of Monte Carlo Simulation has been validated for generating probabilistic forecasts and assessing market risk, as demonstrated by Xiang et al. [5] [6], who investigated its effectiveness in predicting future stock prices. Additionally, evolutionary algorithms have been recognized for their role in optimizing parameters and features in predictive models. Techniques such as Genetic Algorithms and Colony Optimization have been applied to enhance forecasting performance, with a comprehensive review by Kumbure et al. [7] comparing various ML techniques for stock market prediction.

Extensive research on traditional ARIMA (Ariyo et al. [8]) and Facebook's Prophet (Taylor and Letham [9]) models highlights the strengths and weaknesses of different AI approaches for forecasting. Comparative analyses show no single method consistently outperforms others, leading to interest in hybrid models for better capturing financial market complexities.

## III. Methods

The study outlines the methods employed for stock price prediction using conventional Artificial Intelligence (AI) approaches. It is organized into three main aspects: **statistical-stochastic methods, metaheuristic algorithms, and agentic trade strategy tradebot algorithms**. Additionally, for comparison, well-known Machine Learning (ML) approaches (**ARIMA & Prophet**).

### A. *Statistical & Stochastic Methods*

Three distinct statistical-stochastic methods were applied for the problem: **Hidden Markov Models (HMMs), Monte Carlo Simulation**, and **Generalized Autoregressive Conditional Heteroskedasticity (GARCH)** algorithms.

*1) **Hidden Markov Models (HMMs)**:* The HMM approach was implemented for both daily and weekly stock price prediction. Time series data were preprocessed for feature extraction, ensuring that no leakage occurred during the 80% training and 20% testing split.

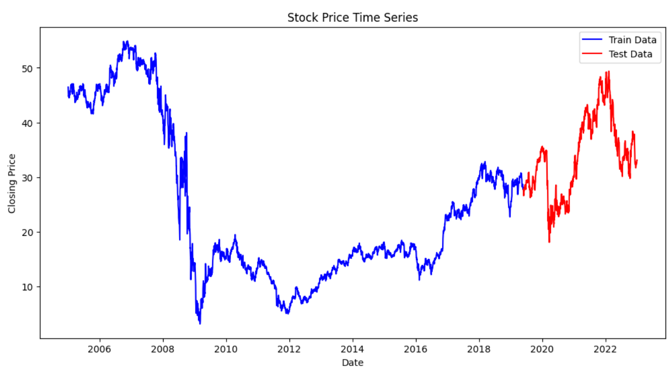


Fig. 1. Splitted Time Series Data of BAC Stock

Two feature configurations were considered:

- **Without Indicators:** 'Returns', 'Volume Change', 'High Low Difference', and 'Open Close Difference' features are used as HMM model inputs.
- **With Indicators:** The above features augmented with technical indicators ('SMA 10', 'RSI', and 'MACD').

For each feature configuration and prediction horizon (**daily** and **weekly**), three HMM models were trained with different numbers of hidden states (2, 3, and 4) to assess performance variations.

Additionally, **Gaussian Mixture Model HMMs (GMMHMMs)** were evaluated under the same conditions. For the GMMHMMs, models were trained with 2, 3, and 4 mixtures per state (n mix), again for both feature configurations and prediction horizons. Model performance was evaluated using accuracy, precision, recall, and f1 score metrics.


Fig. 2. Example HMM with Daily Prediction Setup and 2 Hidden States

*2) **Monte Carlo Simulation**:* Monte Carlo Simulation was employed to generate probabilistic forecasts for daily returns. Daily returns were calculated and scaled, and a normal distribution was assumed based on the mean and standard deviation of the training returns. A large number of simulations (**10,000**) were conducted using the 80% training data, and predictions were tested on the 20% test data. The performance of the Monte Carlo Simulation was assessed using Directional Accuracy, Mean Squared Error (MSE), and Mean Absolute Error (MAE).

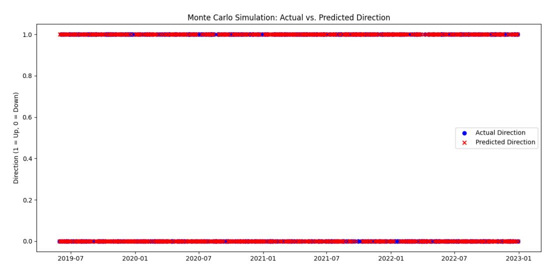


Fig. 3. Monte Carlo Simulations of the Time Series Data

*3) **GARCH Models**:* GARCH models were implemented to forecast daily and weekly returns, using the same train-test split as the other methods. For each prediction scenario, models were configured with three sets of hyperparameters: (p, q) values of (1, 1), (1, 2), and (2, 1). Model performance was evaluated with Directional Accuracy, MAE, and MSE metrics.

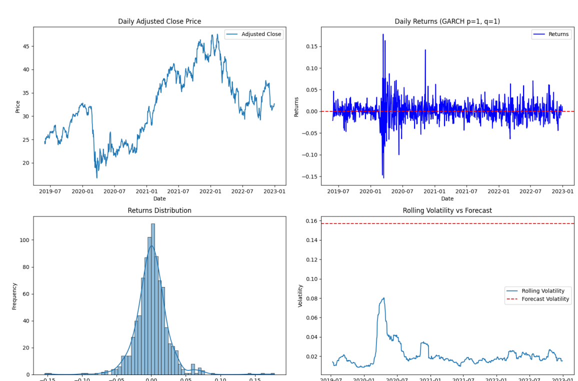


Fig. 4. GARCH Model Findings with Setup p=1&q=1

### B. Metaheuristic Methods

Metaheuristic optimization techniques were used to fine-tune model parameters for enhanced prediction. Three methods were explored: Genetic Algorithm (GA) and Colony Optimization (CO).

*1) Genetic Algorithm (GA):* For GA, a daily directional prediction setup was used. Five models were created by varying key hyperparameters: population size (50 and 100), generations (100 and 200), mutation rate (0.1 and 0.2), and crossover rate (0.7 and 0.9), with a constant tournament size of 3. These models were evaluated on the test set using Directional Accuracy, MSE, and MAE.

*2) Colony Optimization (CO):* In the CO approach, daily returns were the basis for prediction. Five hyperparameter configurations were tested, varying archive size (20 and 30), number of ants (50 and 100), maximum iterations (100 and 200), and pheromone evaporation rate (q) (0.5 and 0.7), while keeping the influence of the best solution (xi) constant at 0.85. The resulting models were evaluated using Directional Accuracy, MSE, and MAE.

### C. Tradebot Agent Strategies

To evaluate the effectiveness of algorithmic trading strategies in stock price prediction, 13 distinct tradebot agents were implemented, each utilizing a unique buy-sell strategy. These tradebots were tested on the historical time series data of BAC stock, allowing for a comparative analysis of their performance under real market conditions.

For all tradebot simulations and backtesting procedures, an initial investment of $10,000 was assumed. Each transaction, including buying and selling, incurred a fixed $1 transaction cost to simulate realistic market conditions. This framework provided a structured approach to analyzing the effectiveness of different trading strategies, offering insights into their profitability, risk exposure, and market adaptability. The strategies employed by the tradebots include:

- **1) Buy and Hold Strategy:** A passive investment approach where the stock is purchased and held without further transactions.
- **2) Monthly Trading Strategy:** Executes trades at the beginning of each month based on predefined conditions.
- **3) "Sell in May and Go Away" Strategy:** Implements a seasonal trading approach where stocks are sold in May and repurchased later in the year.
- **4) Daily Momentum-Based Strategy:** Buys stocks that exhibit upward momentum and sells when momentum weakens.
- **5) Daily Reversal-Based Strategy:** Trades stocks based on expected price reversals from overbought or oversold conditions.
- **6) Moving Average Crossover Strategy (200-day vs. 50-day):** Buys when the short-term moving average crosses above the long-term moving average and sells when the opposite occurs.
- **7) Moving Average Crossover Strategy (50-day vs. 20-day):** A similar strategy with shorter moving average windows to capture medium-term trends.
- **8) Relative Strength Index (RSI) Strategy:** Buys when RSI indicates an oversold condition and sells when it indicates overbought conditions.
- **9) RSI Reentry Strategy:** Adjusts buy-sell decisions based on RSI recovery signals after extreme levels.
- **10) Consecutive Down Days Strategy:** Purchases stock after a predefined number of consecutive declining days, anticipating a reversal.
- **11) Moving Average Crossover Strategy (10-day vs. 5-day):** Uses shorter moving average windows to identify short-term trends.
- **12) 3% Below and Above 50-day Moving Average Strategy:** Executes trades when the stock price deviates significantly from its 50-day moving average.
- **13) Linear Regression-Based Strategy:** Uses regression modeling to predict price movements and determine buy-sell decisions.

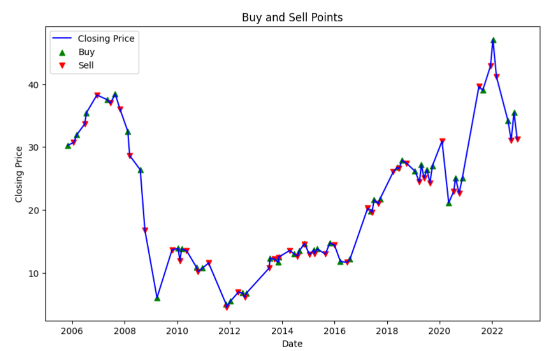


Fig. 5. Strategy 7's buy & sell points

Each tradebot's performance was evaluated using nine frequency trading-oriented metrics:

- **Annualized Return** – Measures the total return on investment over a year.
- **Average Number of Trades in a Year** – Captures the trading frequency of each strategy.
- **Average Trading Profit** – Assesses profitability per trade, which can be negative.
- **Average Transaction Length (Days)** – Evaluates the average duration of each trade.
- **Maximum Loss in a Transaction** – Identifies the highest loss sustained in a single trade.
- **Maximum Profit in a Transaction** – Determines the highest gain from a single trade.
- **Portfolio Maximum Value** – Tracks the highest portfolio value achieved during the simulation.
- **Portfolio Minimum Value** – Identifies the lowest recorded portfolio value.
- **Portfolio Final Value** – Represents the total capital at the end of the simulation.

### D. *Machine Learning (ML) Approaches*

To compare the effectiveness of conventional AI methods with machine learning (ML) models in stock price prediction, two widely used ML models, **AutoRegressive Integrated Moving Average (ARIMA)** and **Prophet**, were implemented. These models were selected due to their strong time series forecasting capabilities and widespread application in financial market predictions.

*1) **ARIMA Model**:* The ARIMA model was applied using the same dataset split as in previous experiments, with 80% used for training and 20% for testing. Prior to model training, the **Augmented Dickey-Fuller (ADF) [10]** test was conducted to determine whether the time series data was stationary or non-stationary. The test result indicated **non-stationarity**, with a p-value of 0.6019, necessitating differencing to achieve stationarity before training the ARIMA model.

To determine optimal hyperparameter values for the model, a Grid Search methodology was employed. The parameters p, d, and q were explored within the ranges:

- **p** (autoregressive lag order): $\{1, 2, 3\}$
- **d** (degree of differencing): $\{0, 1\}$
- **q** (moving average lag order): $\{1, 2, 3\}$

A total of **18 ARIMA models** were trained with different parameter configurations, all focused on daily stock price prediction. The models were evaluated using Accuracy, Root Mean Squared Error (RMSE), and Mean Absolute Error (MAE) metrics.

*2) **Prophet Model**:* The Prophet model was also trained using the same dataset split and was applied for daily stock price forecasting. Six different configurations were tested to assess the model's adaptability under various conditions. These configurations included:

- **Default Configuration** – Standard model settings.
- **Less Flexible Trend** – Lower changepoint_prior_scale value to reduce trend variability.
- **More Flexible Trend** – Higher changepoint_prior_scale value to allow greater trend fluctuations.
- **Less Seasonal Variation** – Lower seasonality_prior_scale value to limit seasonal effects.
- **More Seasonal Variation** – Higher seasonality_prior_scale value to enhance seasonal effects.
- **Less Holiday Impact** – Lower holidays_prior_scale value to reduce the influence of holidays on the forecast.

Each of these six configurations was trained separately to examine the impact of different hyperparameters on forecasting performance. The results were evaluated using Accuracy, Mean Squared Error (MSE), Mean Absolute Error (MAE), and Mean Directional Accuracy Percentage Error (MDAPE) metrics.

## IV. Evaluation

The performance of the conventional Artificial Intelligence (AI) methods, encompassing statistical-stochastic techniques, metaheuristic algorithms, and agentic trade strategy tradebot algorithms, alongside the well-known Machine Learning (ML) approaches, was rigorously assessed using a set of established financial and statistical metrics.

Without indicator features, HMMs trained on weekly data generally outperformed daily models, with the 4-state weekly model achieving the highest accuracy of 60.64%. Including indicator features had a varied impact; the best performing daily HMM reached 55.65% accuracy with 3 states, while the top weekly model achieved 61.20% with 4 states. Detailed analysis of daily HMMs without indicators showed that increasing the number of states from 2 to 4 led to a gradual improvement in test accuracy, ranging from 51.21% to 53.75%.

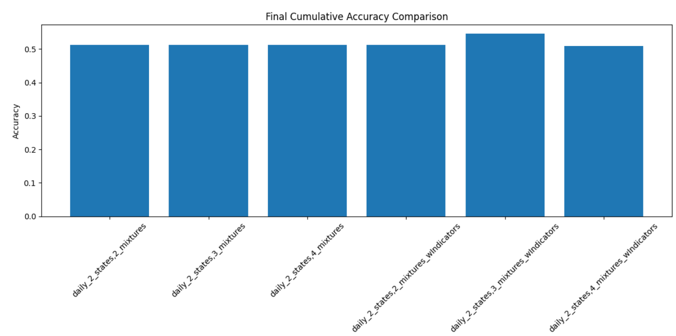


Fig. 6. Best Performing HMM Model Accuracies

Next, Gaussian Mixture Model Hidden Markov Models (GMMHMMs) were evaluated. Without indicator features, daily GMMHMMs showed a consistent accuracy of 51.21% across different numbers of states. However, with indicator features, the accuracy of the 2-state GMMHMM varied with the number of mixtures, with the 3-mixture model achieving the highest accuracy of 54.55%. Detailed evaluation of these models revealed similar performance in terms of accuracy, precision, recall, and F1-score, with an upward prediction trend and comparable rolling accuracy ranges.

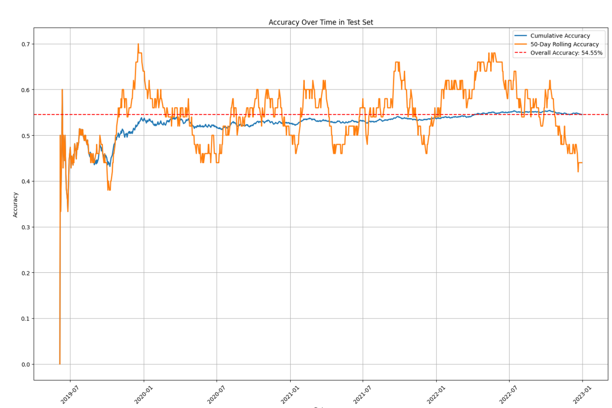


Fig. 7. Best Performing Model's Cumulative Accuracy over Test Data

The Monte Carlo Simulation approach, based on 10,000 simulations, achieved a directional accuracy of **52.10%** with both the Mean Squared Error (MSE) and Mean Absolute Error (MAE) equal to 0.4790. This method provides a probabilistic forecast that, while modest in directional accuracy, offers a stable estimation of error metrics.

GARCH models consistently predicted a downward trend (60% confidence) but showed low directional accuracy: around **48.79%** for daily and **43.09%** for weekly predictions, with higher errors for weekly data. **Their directional prediction ability was below the no-learning threshold, making them ineffective for this scenario despite capturing volatility.**

Metaheuristic approaches were assessed by running multiple experiments with varying hyperparameters. In the Genetic Algorithm (GA) experiments, training accuracies consistently reached around 53% with test directional accuracies ranging from approximately 49% to 51%, and corresponding MSE and MAE values close to 0.49–0.51. Notably, configurations with a population size of 50 and 100 generations, or with slightly altered mutation or crossover rates, produced comparable results, with the highest directional accuracy observed at about 51.38%.

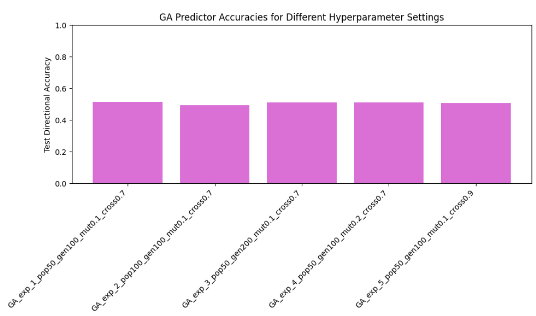


Fig. 8. Genetic Algorithm Experiment Results

Similarly, the Colony Optimization (CO) experiments demonstrated stable performance, with test directional accuracies varying narrowly between 50.28% and 50.83% and training fitness values around 53%. The error metrics for CO were also similar across configurations, with MSE and MAE values near 0.49–0.50.

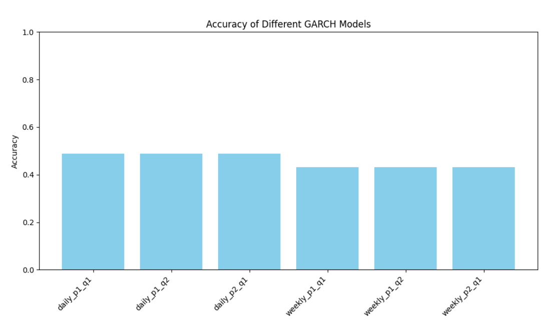


Fig. 9. Ant Colony Optimization Experiment Results

Overall, both metaheuristic methods yielded comparable forecasting performance, with GA achieving marginally higher directional accuracy in some configurations. These results indicate that while both approaches offer robust parameter optimization, the improvements in directional prediction remain modest.

The evaluation of 13 tradebot agents showed that the best strategy depends on individual investor preferences for risk, trading frequency, and volatility tolerance. While no single strategy was universally superior, Strategy 7, a 50/20-day moving average crossover, stood out for long-term investors due to its highest profitability (341%) and low commission costs from fewer trades. However, this strategy involves significant portfolio volatility, with potential losses and gains reaching thousands of dollars per trade, making it most suitable for investors with a high risk appetite comfortable with substantial short-term fluctuations.

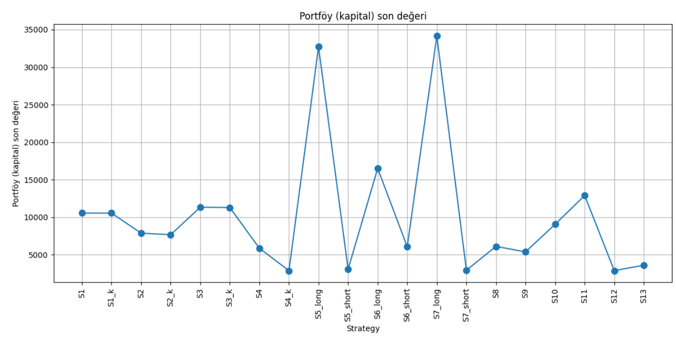


Fig. 10. Portfolio Values of Each Strategy after Simulation

Machine learning approaches were evaluated using ARIMA and Prophet. ARIMA models, applied to a non-stationary time series, achieved daily directional accuracies ranging from 48% to 51%, with performance varying based on parameter configurations.

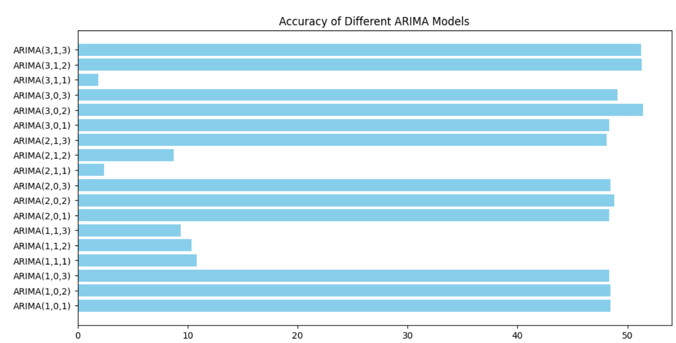


Fig. 11. ARIMA Model Results

In contrast, Prophet consistently achieved around 61.29% accuracy for 1-day predictions, outperforming ARIMA in the very short term. For 7-day predictions, both models showed comparable accuracy (around 50-52%). Overall, Prophet appeared more effective for immediate short-term forecasts, while ARIMA offered more variable but sometimes competitive daily performance. The choice between them depends on the specific forecasting horizon and desired balance between accuracy and error metrics.

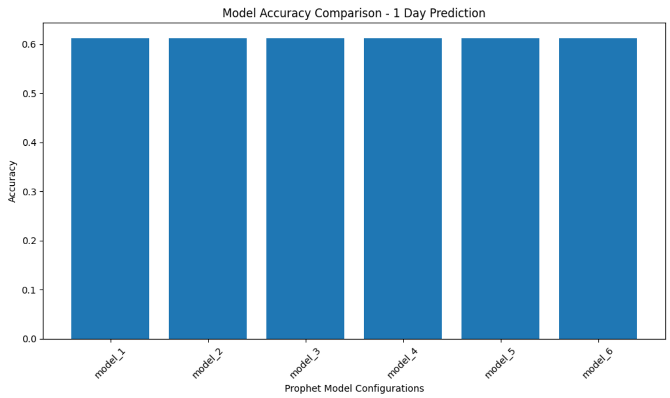


Fig. 12. Prophet Model Results

## V. Conclusion

This study compared conventional AI methods with machine learning models for stock price prediction using BAC time series data from 2005 to 2022. Conventional approaches—including statistical-stochastic techniques (HMM, Monte Carlo Simulation, GARCH), metaheuristic algorithms (Genetic Algorithm, Colony Optimization), and 13 agentic tradebot strategies—demonstrated efficient parameter optimization and stable performance, often with lower computational overhead. For example, weekly HMM models achieved accuracies above 60%, and metaheuristic methods consistently delivered directional accuracies around 51%.

In contrast, machine learning models, such as ARIMA and Prophet, while computationally more demanding, proved crucial for improving prediction accuracy. ARIMA models produced daily directional accuracies between 48% and 51%, with performance varying by configuration. Prophet, however, consistently achieved about 61% accuracy for 1-day forecasts and comparable results for longer horizons, highlighting its strength in very short-term predictions.

Overall, the findings indicate that conventional AI methods offer efficiency and robust performance with lower complexity, making them attractive for scenarios where computational cost is a concern. Meanwhile, ML models excel in accuracy and can be vital for capturing rapid market changes, particularly in the short term. The optimal approach ultimately depends on the investor's risk tolerance, trading frequency, and desired balance between efficiency and predictive accuracy. Future work may explore hybrid models that integrate the efficiency of conventional methods with the superior accuracy of ML techniques to better capture complex market dynamics. The code is available on GitHub repository: GitHub

## References


[1] Thio-ac, August  Cabrales, Daniel  Calibara, Drexanne  Madrazo, Maries  Matawaran, Agustine  Micaller, Genesis  Fernandez, Edmon Amado, Timothy  Jorda, Romeo Jr  Tolentino, Lean Karlo  Enriquez, Lejan Alfred. (2023). Stock Price Prediction and Portfolio Optimization Using Hidden Markov Model and Quadratic Programming.

[2] Mutinda, John  Ph.D., Amos. (2024). Stock price prediction using combined GARCH-AI models. Scientific African. 26. e02374.

[3] Tupe-Waghmare, Priyanka. (2021). Prediction of Stocks and Stock Price using Artificial Intelligence : A Bibliometric Study using Scopus Database.

[4] Omer Berat Sezer, Mehmet Ugur Gudelek, Ahmet Murat Ozbayoglu, Financial time series forecasting with deep learning : A systematic literature review: 2005–2019, Applied Soft Computing

[5] Xiang, Jeremy Ng Phak et al. "Monte Carlo Simulation Prediction of Stock Prices." 2021 14th International Conference on Developments in eSystems Engineering (DeSE) (2021): 212-216.

[6] Xiang, Jeremy Ng Phak et al. "Monte Carlo Simulation Prediction of Stock Prices." 2021 14th International Conference on Developments in eSystems Engineering (DeSE) (2021): 212-216.

[7] Mahinda Mailagaha Kumbure, Christoph Lohrmann, Pasi Luukka, Jari Porras, Machine learning techniques and data for stock market forecasting: A literature review, Expert Systems with Applications

[8] A. A. Ariyo, A. O. Adewumi and C. K. Ayo, "Stock Price Prediction Using the ARIMA Model," 2014 UKSim-AMSS 16th International Conference on Computer Modelling and Simulation, Cambridge, UK, 2014, pp. 106-112, doi: 10.1109/UKSim.2014.67.

[9] Taylor, S. J.,  Letham, B. (2018). Forecasting at Scale. The American Statistician, 72(1), 37–45.

[10] Dickey, D.  Fuller, Wayne. (1979). Distribution of the Estimators for Autoregressive Time Series With a Unit Root. JASA. Journal of the American Statistical Association. 74. 10.2307/2286348.