

Algorithmic Trade Bot

B Gnaneshwari
CSE dept.

Rajiv Gandhi Institute of
Technology
Bangalore, India
gnana0128@gmail.com

Nikitha G
CSE dept.

Rajiv Gandhi Institute of
Technology
Bangalore, India
nikydv3@gmail.com

Subhiksha S T
CSE dept.

Rajiv Gandhi Institute of
Technology
Bangalore, India
subhisubhiksha8@gmail.com

Tanishka V
CSE dept.
Rajiv Gandhi Institute of
Technology
Bangalore, India
tanishkav008@gmail.com

Mrs. Kavyashree J Assistant
Professor Rajiv Gandhi
Institute of Technology
Bangalore, India
Kavyacse.rgit@gmail.com

Abstract— Algorithms that follow a trend and a predetermined set of rules are used in algorithmic trading to execute trades. Predictive Trade Bot systems aim to provide unusual profits for stock order, position, balance, sell, and by tools. The trade can produce income at an unnaturally high speed and frequency. By developing an Algorithmic Trading Bot that will automatically trade user strategies along with its own algorithms for day-to-day trading based on different market conditions and user approach, and throughout the course of the day invest and trade with continuous modifications to ensure the best trading results, our project seeks to advance this revolution in the markets of the future. Algorithmic Trading Bot's primary goal is to minimize the risk of mistakes in trading performed by humans.

Keywords— Algorithmic trading, Finance, Random Forest, coin market cap, yahoo API.

INTRODUCTION

The financial markets are subject to fluctuations that can make them risky for investors, but there are strategies that can be employed to generate profits, such as identifying companies with rapid earnings growth. Investors can choose between passive and active investment strategies, with the former being a simple and low-cost way to invest in the market while the latter involves more frequent trading and individual stock selection. While speculation can help balance supply and demand and lead to price equilibrium, it can also be risky and may not always reflect the true state of the economy. As such, investors should carefully consider their investment goals and strategies, remain informed about market trends, and monitor their investments regularly to ensure they align with their objectives.

The strategy involves purchasing stocks when their price is below the average value and selling them when the price exceeds that value. By studying and analyzing past trends, investors can sometimes make predictions about future market movements. Speculators are currently viewed as a source of information on market conditions, and their techniques are continuously evolving. There are two commonly used approaches to speculation: a traditional approach based on technical indicators used in econometrics.

There are various investment strategies used in the Forex market, including day trading, trading news, and technical indicator trading using the SMTP protocol. The majority of Trade market investors do not manually trade, but instead utilize computer algorithms to implement simple or complex strategies. By using an API application system, the client can monitor the trading bot and current stock market conditions in real-time. This system allows for easy communication and integration of various application and system components on an internal network.

The following are some of the objectives related to trading bots and the trading market:

It is anticipated that the global trading market will experience substantial growth from 2018 to 2026. Trading bots utilize the SMTP protocol for stock market trading and send information to the client. Programmed software allows crypto bots to purchase, sell, and retain digital assets. APIs enable communication between applications and system components on both internal networks and Internet. The use of trading bots can help reduce exchange costs.

LITERATURE SURVEY

[1]. "The Rise of the Machines: Algorithmic Trading in the Foreign Exchange Market" by Alain Chaboud, Benjamin Chiquoine, Erik Hjalmarsson, and Clara Vega. This paper examines the impact of algorithmic trading on liquidity and volatility in the foreign exchange market.

The authors use a unique dataset of all transactions in the EUR/USD spot market for the period from January 2002 to December 2004. They find that algorithmic trading accounts for a significant and growing proportion of trading volume, particularly during periods of high volatility. The paper also shows that algorithmic trading has a positive impact on liquidity in the foreign exchange market. Specifically, the authors find that algorithmic traders provide liquidity by placing limit orders in the order book, which reduces bid-ask spreads and improves market efficiency.

[2]. "Machine Learning for Trading: A Literature Survey" is an academic paper written by Stefan Zohren and Stephen Roberts, and published in the IEEE Transactions on Neural Networks and Learning Systems in 2018. The paper provides a comprehensive survey of machine learning techniques used in trading and investment.

The authors review a wide range of topics related to machine learning for trading, including data sources, feature extraction, prediction models, portfolio optimization, and risk management. They also discuss the challenges associated with using machine learning in trading, such as overfitting, data snooping bias, and lack of interpretability. The paper presents an overview of the various machine learning techniques that have been used in trading, including supervised learning, unsupervised learning, reinforcement learning, and deep learning. The authors provide examples of how these techniques have been used to predict market movements, identify trading opportunities, and optimize portfolios.

[3]. "The Effect of Algorithmic Trading on Liquidity and Price Efficiency" is an academic paper written by Terrence Hendershott, Charles M. Jones, and Albert J. Menkveld, and published in the Journal of Finance in 2011. The paper investigates the impact of algorithmic trading on liquidity and price efficiency in financial markets.

The authors analyze a large dataset of algorithmic trades in the S&P 500 futures market, and find that algorithmic trading is associated with higher levels of liquidity, measured by narrower bid-ask spreads and

higher trading volume. They also find that algorithmic trading improves price efficiency, as measured by a lower correlation between order flow and subsequent price changes. The paper provides evidence that algorithmic trading contributes to the overall efficiency of financial markets by improving liquidity and price discovery. However, the authors also note that the use of algorithmic trading can lead to increased volatility, particularly during periods of market stress.

[4]. "A Deep Reinforcement Learning Framework for the Financial Portfolio Management Problem" is an academic paper by Shuai Zhao and Huimin Zhao. The paper proposes a deep reinforcement learning framework for portfolio management. The framework uses a neural network to learn a trading policy that maximizes portfolio returns while minimizing risk.

The environment is modeled as a Markov Decision Process (MDP), where the state of the system is represented by a vector of asset prices and portfolio weights. The agent is represented by a deep neural network that maps the state of the system to a distribution over portfolio weights. The neural network is trained using reinforcement learning to maximize the expected return of the portfolio while minimizing the risk. The agent is trained using a variant of the deep Q-learning algorithm called the deep deterministic policy gradient (DDPG) algorithm. The DDPG algorithm uses a replay buffer to store past experiences and updates the agent's neural network parameters using a combination of gradient ascent and gradient descent.

The paper presents experimental results that demonstrate the effectiveness of the proposed deep reinforcement learning framework for portfolio management. The experiments show that the framework outperforms traditional portfolio management methods such as the mean-variance optimization and the risk parity method. The paper also discusses the challenges and limitations of using deep reinforcement learning for portfolio management, such as the need for large amounts of data and the potential for overfitting.

[5]. "Algorithmic Trading and Market Dynamics" is an academic paper by Albert J. Menkveld. The paper investigates the impact of algorithmic trading on market dynamics, including liquidity, volatility, and price efficiency.

The paper provides a comprehensive review of the literature on algorithmic trading and its impact on financial markets. It discusses the advantages of algorithmic trading, such as improved liquidity, price discovery, and execution efficiency.

PROPOSED METHODOLOGY

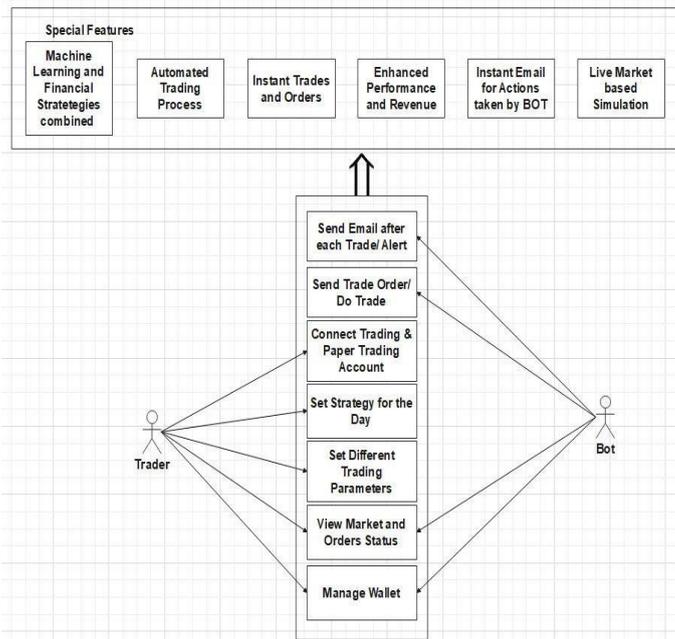


Figure 1: Architecture of Algorithmic Trading Bot

Our proposed solution's architectural diagram illustrates two distinct roles: Trader and Bot. The Trader can access trade orders, market statistics, account management, and set up a day trading strategy through the bot. On the other hand, the Bot validates and executes trades based on market and user statistics, sends notifications, and has access to the user wallet to execute trade orders. The diagram also highlights several unique features at the top.

Data Pre-processing

Data pre-processing is applied on the dataset to get intraday movements to pass into Random Forest Regressor.

1. We drop all other columns except Date and Close price.
2. To determine the actual trading signal, we assume that we traded on a prior day's close price, this is done by lagging the data by 1 day. We create a lag for 41 days.
3. We then clean the dataframe by dropping any NULL values.
4. Dataset is split as [0:33] data into X (inputs) and the rest into Y (outputs)

Splitting dataset into Test and Train dataset

The dataset is divided into training and testing sets with a ratio of 60:40. This results in the creation of four variables, namely X_train, X_test (for inputs), and Y_train, Y_test (for outputs).

Training the Random Forest Regression model on the training set

The Random Forest Regressor class is imported and assigned to the variable regressor. The .fit() function is then used to fit the training input and output values (X_train and Y_train) to the regressor

by reshaping it. The feature importance is calculated using the regressor feature importance method to evaluate the significance of selected features and enhance the model's performance.

Predicting the Results

We predict the results of the test set with the model trained on the training set values using the regressor.predict function and assign it to 'Y_predicted'.

Visualizing the Random Forest Regression Results

A graph of Share Price (Both Y_test & Y_predicted) vs Date is plotted. The Actual values are plotted with "Red" color and the Predicted values with "Blue" color.

Integration of Financial Strategy Bot with Random Forest Model

A Python Bot is developed that links to a Paper Trading account through API. The user inputs the strategy parameters, and once trading starts, the Bot operates until the Stop Loss is triggered, the Market is closed, or the User sends a Stop signal to the Bot.

The Bot monitors the Market conditions and current Positions to determine its action. The RandomForest model is included in the Bot as a job lib file, and the Bot makes decisions based on the model's prediction and the financial strategy.

MODULES INVOLVED:

Data feed module: This module fetches data from various sources such as exchanges, news outlets, and social media platforms. It can use APIs, web scraping techniques, or other methods to gather data.

Data processing module: Once the data is collected, it needs to be processed and analyzed to identify patterns, trends, and other useful information. This module can use statistical analysis, machine learning, or other techniques to extract insights from the data.

Trading strategy module: Based on the insights obtained from the data, the trading strategy module can generate buy/sell signals or execute trades automatically. This module can use various algorithms such as moving averages, momentum indicators, and pattern recognition to make trading decisions.

Order management module: This module handles the execution of trades and manages the portfolio of assets. It can interact with the exchange APIs to place orders, track positions, and manage risk.

Performance monitoring module: To evaluate the performance of the trading bot, a monitoring module can track various metrics such as profit/loss, win rate,

and drawdown. It can generate reports and alerts to notify the user of any significant changes in performance.

User interface module: A user interface module can provide a dashboard or other graphical interface to allow the user to monitor and control the bot's behavior. It can also display real-time data and analytics to help the user make informed decisions.

ALGORITHMS

Using only Random Forest Algorithm

For over 70 years, seasonal impacts and normal patterns in financial data have been extensively documented in the field of financial economics. This approach suggests using an expert system that employs innovative artificial intelligence techniques to forecast price returns during these seasonal periods. The predictions are then used to develop a profitable trading strategy.

The authors suggest a computerized trading system that utilizes performance-weighted collections of random forests to increase profitability and stability during times of volatility.

The study examines various regression methods and explores the advantages of different expert weighting techniques. The findings reveal that recent-weighted ensembles of random forests yield superior results in terms of both profitability and prediction accuracy compared to other ensemble methods.

Using Random Forest and Probit Regression

In the Forex market, there are many currency pairs and traders, and each pair and individual has their own unique characteristics. Finding the best trading strategy is a complex task. The authors propose a prediction and decision model that generates a profitable intraweek investment strategy. This approach improves trading outcomes for high-frequency intraweek trading. The findings suggest that combining multiple algorithms can improve Forex portfolio management.

The use of a combination of sequence and Probit regression in algorithmic trading is believed to be effective in improving prediction accuracy. This combination helps to identify favorable times to buy or sell currency pairs.

The second figure illustrates the assessment of the Random Forest's performance, presented as a plot displaying both actual and predicted values.

The third figure in this paper displays the evaluation of the performance of Probit Regression.

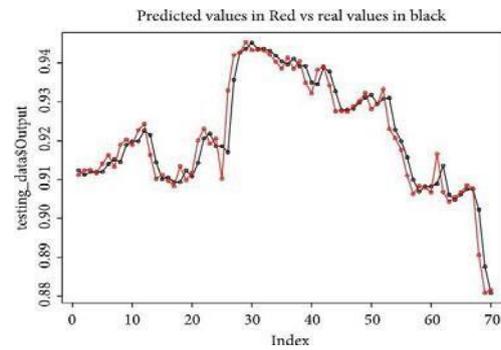


Figure 2: The predicted values and real values are compared in a graph, with the predicted values shown in red and the real values in black. This evaluation is for the Random Forest regression using 500 trees and testing 8 variables for each split.

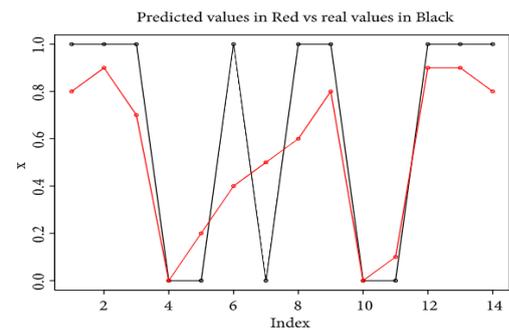


Figure 3: This graph compares predicted values (in red) with real values (in black) for Probit regression.

DATASET

The Coin market cap API and Yahoo Finance are utilized to retrieve historical data, which is then compiled into a dataset. The dataset includes the date, opening price, highest price, lowest price, closing price, and volume traded for each specific day of a particular stock.

Database Splitting

The dataset is divided into two parts, with a 60:40 split ratio. From these two parts, four variables are generated: X_train and X_test, which represent the input data for training and testing respectively, and Y_train and Y_test, which represent the output data for training and testing respectively.

Annotation Description

The dataset includes several columns such as Date, Open Price, High Price, Low Price, Close Price, and Volume traded for a particular stock on a daily basis. However, for our Random Forest Regressor and prediction, we only need Date and Close Price columns for the specific stock. Close Prices will assist us in determining the trend or Moving Average for our Intraday trading of that particular stock. This will be

combined with financial strategies to improve performance with greater accuracy utilizing the predictive power of the Random Forest Regressor.

RESULT

Evaluation based on Metrics –

Below figure shows the performance of our model against the evaluation parameters discussed earlier.

Random Forest Regressor Model:

Random Forest Regressor Model for Trading Analysis –

(Red: Actual Stock Price Movement, Blue: Bot predicted Stock Price Movement)

X axis: 'TSLA' Stock share prices of test dataset and predicted prices

Y axis: Dates of trade

The figure 4 presents a graph where the Share Price is plotted against the Date, showcasing the performance of the Random Forest Model.

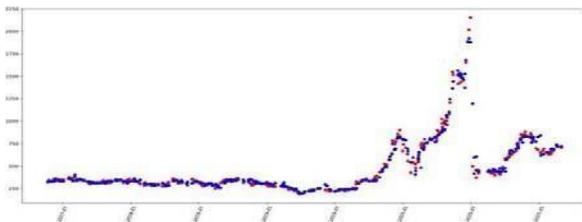


Figure 4: Share Price vs Date Graph

CONCLUSION

The use of Algorithmic Trading Bots is not only beneficial in terms of security, cost, and speed, but it is also a transformative technology for the financial markets and economy. These bots make it easier for both novice and experienced traders to achieve profitable outcomes with reduced effort, time, and losses. The combination of financial knowledge with machine learning is becoming increasingly important in future trading as it can enhance both performance and revenue.

REFERENCES

[1] Younes Chihab, Zineb Bousbaa, Marouane Chihab, Omar Bencharef, Soumia Ziti, and Miin-Shen Yang. 2019. Algo-Trading Strategy for Intra-week Foreign Exchange Speculation Based on Random Forest and Probit Regression. *Appl. Comp. Intell. Soft*

Comput. 2019(2019).DOI:

<https://doi.org/10.1155/2019/8342461>

[2] Suryoday Basak, Saibal Kar, Snehanu Saha, Luckyson Khaidem, Sudeepa Roy Dey, Predicting the direction of stock market prices using tree-based classifiers, *The North American Journal of Economics and Finance*, Volume 47, 2019, Pages 552- 567, ISSN1062-9408, <https://doi.org/10.1016/j.najef.2018.06.013>.

[3] Bruno Miranda Henrique, Vinicius Amorim Sobreiro, Herbert Kimura, Stock price prediction using support vector regression on daily and up to the minute prices, *The Journal of Finance and Data Science*, Volume 4, Issue 3, 2018, Pages 183- 201, ISSN2405-9188, <https://doi.org/10.1016/j.jfds.2018.04.003>.

[4] Omer Berat Sezer, Murat Ozbayoglu, Erdogan Dogdu, A Deep Neural-Network Based Stock Trading System Based on Evolutionary Optimized Technical Analysis Parameters, *Procedia Computer Science*, Volume 114, 2017, Pages 473-480, ISSN 1877-0509, <https://doi.org/10.1016/j.procs.2017.09.031>.

[5] Manoj Thakur and Deepak Kumar. 2018. A hybrid financial trading support system using multi-category classifiers and random forest. *Appl. Soft Comput.* 67, C (June 2018), 337–349. DOI: <https://doi.org/10.1016/j.asoc.2018.03.006>

[6] M. Nabipour, P. Nayyeri, H. Jabani, S. S. and A. Mosavi, "Predicting Stock Market Trends Using Machine Learning and Deep Learning Algorithms Via Continuous and Binary Data; a Comparative Analysis," in *IEEE Access*, vol. 8, pp. 150199- 150212, 2020, doi: 10.1109/ACCESS.2020.3015966.

[7] Viliam Vajda, Could a Trader Using Only "Old" Technical Indicator be Successful at the Forex Market?, *Procedia Economics and Finance*, Volume 15, 2014, Pages 318-325, ISSN 2212- 5671, [https://doi.org/10.1016/S2212-5671\(14\)00515-2](https://doi.org/10.1016/S2212-5671(14)00515-2).

[8] Wang, J., Wu, X. and Zhang, C., 2005. Support vector machines based on K-means clustering for real-time business intelligence systems. *International Journal of Business Intelligence and Data Mining*, 1(1), pp.54-64. Google Scholar