International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 07 Issue: 11 | November - 2023 SJIF Rating: 8.176 ISSN: 2582-3930

Machine Learning-Based Momentum Strategy for High Frequency Trading and Buy/Sell Calls Using Algorithmic Trading

Anupriya Dasgupta
Electronics and Telecommunications
Sardar Patel Institute of Technology
Mumbai, India
anupriya.dasgupta@spit.ac.in

Oindrella Ghosh
Electronics and Telecommunications
Sardar Patel Institute of Technology
Mumbai, India
oindrella.ghosh@spit.ac.in

Sanskar Laddha
Electronics and Telecommunications
Sardar Patel Institute of Technology
Mumbai, India
sanskar.laddha@spit.ac.in

Milind Paraye
Electronics and Telecommunications
Sardar Patel Institute of Technology
Mumbai, India
milind_paraye@spit.ac.in

***Abstract* — Stock Market Prediction is beneficial to investors. It provides shareholders with useful assistance in making suitable decisions about whether to purchase or sell shares. Accurate stock price prediction is extremely challenging because of multiple factors. Stock prices are influenced by a variety of factors, but the price at any given time is determined by supply and demand in the market. Due to the growing volume of data, it is now impractical, if not impossible for humans to manually analyze data for certain tasks like predicting stock market movements, necessitating automation. The stock price data represents a financial time series data which becomes more difficult to predict due to its characteristics and dynamic nature. Thus, this also means that there is a lot of data to find patterns in. This gives rise to the concept of algorithmic trading, which uses automated, pre-programmed trading strategies to execute orders. Machine learning algorithms explore large amounts of data and search for a model that will achieve the programmer's goal. Our project's goal is to use machine learning to implement a momentum strategy. We attempted to predict trading signals using machine learning techniques based on a set of technical indicators and rules.**

I. INTRODUCTION

Algo Trading is a framework that allows users to build bots capable of autonomous trading, as well as develop and test their strategies. Although designed primarily for crypto markets, it can also be modified to run in other markets. It applies the same principle to all three operating modules: real-time, backtest, and tick-by-tick. It runs indefinitely, reading the input data and making decisions based on it. Working synchronously at the moment, waiting for data to take action, which could be to buy, sell, or do nothing based on previously available data. In order to open and close trades based on computer code, algorithmic trading combines financial markets and software. When trades are opened or closed can be decided by investors and traders. To engage in high-frequency trading, they can also make use of computing power. Algorithmic trading is a common practice in today's financial markets with a wide range of strategies available to traders.

By implementing algorithmic trading, we have built our project to induce buy/sell calls on the basis of how the particular stock taken into consideration is performing. The ability to purchase an underlying stock at a predetermined price up until a given expiration date is known as a call option. A put option, on the other hand, is the right to sell the underlying stock at a predetermined price until a specified expiry date.

A call option buyer has the right (but not the obligation) to buy shares at the strike price before or on the expiry date, whereas a put option buyer has the right (but not the obligation) to sell shares at the strike price. A holder is the person who purchases a call option. A call option is purchased in the hope that the price will rise above the strike price and before the expiration date. The buyer will not exercise the option if the price does not rise above the strike price. The buyer will incur a loss equal to the call option premium. Writers, also referred to as call option sellers, trade call options in the expectation that they will expire worthless. If the option buyer profitably exercises their option when the price of the underlying security rises above the option strike price, their profit will be diminished or may even result in a net loss. An analyst's recommendation to hold a security means not to buy it or sell it. A company with a hold recommendation is typically expected to perform in line with the market or at a similar rate to other companies in its industry.

In a nutshell, algorithmic trading gives investors the ability to make more trades in a shorter amount of time without being affected by human emotions or trading mistakes. The following is an illustration of algorithmic trading instructions:

- Purchase 100 shares of the XYZ Company if the price rises to Rs. 450 before 2:00 PM. Now, the algorithmic trading order will automatically place an order for 100 shares of the XYZ company if the share price rises above Rs 450. The order will only be carried out by the algorithmic trading program, though, if the target price is reached before 2:00 PM. The directives are null and void after 2:00 PM.

- If company QPR's 20-day moving average dips below the 200-day moving average prior to market close, sell 100 shares of the company. In this scenario, if the 20-day moving average of the QPR company dips below the 200-day moving average before the market closes, the algorithmic trading software will sell 100 shares of QPR. Otherwise, the order is not carried out.

Orders must be executed by algorithmic trading only once after the set instructions have been followed. For instance, in the case of the first instruction, the algorithmic trading software will place the buy order if the price rises above Rs 450 for even a brief period of time. It's possible that the price will drop below Rs 450 once more after those brief intervals. However, the order would have already been placed at the current market price or any other price the investor had specified above Rs 450.

High-frequency trading, also known as HFT, is a method of trading that uses powerful computer programs to transact a large number of orders in fractions of a second. It uses complex algorithms to analyze multiple markets and execute orders based on market conditions.

Currency trading or forex trading is to buy or sell currency in pairs. For example, today the US dollar stands at 79.37 Indian rupees – if you expect the dollar to appreciate against the rupee, you buy more dollars. Conversely, if you expect the dollar to depreciate against the rupee, you will buy rupees.

II. LITERATURE REVIEW

As a part of the literature review, we studied nine papers in total. By observing the technical gaps present in them, we devised a faster, efficient and cost effective system.

[1] In the first paper titled 'Stock Market Prediction Using Machine Learning' published in the 2018 First International Conference On Secure Cyber Computing and Communication (ICSCCC) by the authors I. Parmar, Navanshu Agarwal, Sheirsh Saxena, Ridam Arora, Shikhin Gupta, Himanshu Dhiman, Lokesh Chouhan, use of regression and LSTM based Machine learning to predict stock values was observed. Factors considered by them were open, close, low and high volume. However on analyzing the technical gap in this paper was the use of a very small dataset for training the model. Further the factor of news and corresponding sentiment analysis had not been taken into account by the authors.

[2] The next paper that we took up was titled 'Stock Closing Price Prediction Using Machine Learning Techniques' from 'Procedia Computer Science, Volume 167, 2020, Pages 599-606' by the authors Mehar Vijh, Deeksha Chandola, Vinay Anand Tikkiwal, Arun Kumar. This paper utilized artificial neural network and random forest techniques for predicting the next day closing price for five companies belonging to different sectors of operation. However, features of the dataset used to train the model were not sufficient for accurate prediction.

[3] The third paper named 'Short-Term Stock Market Price Trend Prediction Using a Comprehensive Deep-Learning System' from 'Journal of Big Data Vo. 7, Article number, 66 (2020)' was authored by Jingyi Shen & M. Omair Shafiq. The authors pre-processed the stock market dataset, utilized multiple feature engineering techniques and combined with a customized deep-learning based system for stock market price trend prediction. Unfortunately, the algorithm used in their model was not sensitive to the term lengths other than 2-day, weekly, biweekly. Other parameters like news analysis could also have been used for a more comprehensive model.

[4] Moving on to the fourth paper under the title 'Machine Learning Approaches in Stock Price Prediction: A Systematic Review' from 'Journal of Physics: Conference Series, Volume 2161, 1st International Conference on Artificial Intelligence, Computational Electronics and Communication System (AICECS 2021)' was authored by Payal Soni, Yogya Tewari and Deepa Krishnan. They provided detailed analysis of the techniques from traditional machine learning deep learning methods to neural networks and graph-based approaches, employed in predicting the stock prices as well as explored the challenges entailed along with the future scope of work in the domain. However, it did not combine the sentiment analysis of stocks related information and the numeric value associated with the historical value of stocks. Stock recommendation system could also have been built with more extensive research on the domain.

[5] The next paper named 'A Decision Support Approach For Online Stock Forum Sentiment Analysis' published in 'IEEE Transaction on Systems, Man, Cybernetics: Systems, Vol. 44 No 8' by Desheng Dash Wu, Lijuan Zheng, David L. Olson integrated sentiment analysis into machine learning approaches based on support vector machines and generalized auto-regressive conditional heteroskedasticity modeling. Cross-country results with large data could provide some insights as to whether any commonality exists among different sentiment measures across different countries. A more efficient sentiment calculation algorithm could be developed to enhance the accuracy of judging sentiment polarity of stock reviews.

[6] In the sixth paper titled 'A Survey On Stock Market Prediction Using Machine Learning Techniques' from '1st International Conference on Data Science, Machine Learning and Applications (ICDSMLA) 2019' authored by Polamuri, Dr & Srinivas, Kudipudi & Mohan, A presented a review and analysis of different stock market prediction parameter techniques such as the ANN, support vector machine for stock market prediction, Hidden Markov Model, ARIMA, Time Series Linear Model and RNN. However, only an analysis was provided, not the final approach. They planned on combining two or more methods to construct a novel approach method to improve prediction results.

[7]In the seventh paper titled 'Stock Market Prediction Analysis by Incorporating Social and News Opinion and Sentiment' from '2018 IEEE International Conference on

Data Mining Workshops (ICDMW)' authored by Zhaoxia Wang; Seng-Beng Ho; Zhiping Lin an enhanced learning-based method for stock price prediction that considered the effects of news sentiments was given. It used the real stock price dataset to show an improvement of performance in terms of the mean square error (MSE). On observing the technical gap, we inferred that 'How do researchers select the suitable sentiment analysis parameters for solving this prediction problem' is an unsolved issue.

[8] The eighth paper named 'Stock Price Prediction Using News Sentiment Analysis' published in '2019 IEEE Fifth International Conference on Big Data Computing Service and Applications (BigDataService)' authored by S. Mohan, S. Mullapudi, S. Sammeta, P. Vijayvergia and D. C. Anastasiu gathered a large amount of time series data and analysed it in relation to related news articles, using deep learning models to improve the accuracy of stock price predictions. It used cloud-computing for training prediction models and performing inference for a given stock in real-time. However,the models did not perform well in cases where stock prices were highly low or volatile.

[9] Finally, the ninth paper that we took up was titled 'Analysis of the Impact of High-Frequency Trading on Artificial Market Liquidity' from 'IEEE Transactions on Computational Social Systems, vol. 7, no. 6,pp 1324-1334, Dec. 2020' authored by I. Yagi, Y. Masuda and T. Mizuta. Agent-based simulations to compare the major liquidity indicators in an artificial market where an HFT participated was compared to one where no HFT participated. Unfortunately, changes in market liquidity indicators and the relationships between them in markets where the fundamental price was unstable had not been taken into account.

III. DATASET USED

To conduct our study, we obtained daily historical data from Yahoo Finance. We selected four stocks (BIDU, MSFT, AAPL, and TXN) from the technology sector (NDXT index) of the NASDAQ. The time period covered was from January 1, 2007, to December 31, 2017. The dataset comprised six variables: date, opening price of the day, highest price of the day, lowest price of the day, closing price of the day, and traded volume. We divided the data into an 80% training set and a 20% test set.

IV. METHODOLOGY AND IMPLEMENTATION

Our proposed model is divided into two parts based on their functionalities. The first part makes use of historical data to predict stock price movement using five different machine learning algorithms, the best algorithm out of the five is further deployed to general buy/sell/hold calls. The second part of our model is designed in a way such as to develop a strategy to perform real time automated currency trading and additionally backtest the strategy for better results.

In this study, we aimed to implement a momentum strategy using machine learning techniques to predict stock market movements. The increasing volume of data has made manual analysis impractical for tasks like stock market prediction, necessitating automation. Machine learning algorithms offer a solution by exploring large datasets and searching for models that can achieve the desired goals. Our objective was to predict trading signals based on a set of rules using technical indicators.

To address the problem of stock trading decisions, we formulated it as a classification problem with two classes: buy and sell. The goal was to identify the most efficient classifier based on certain metrics. We selected several momentum and volatility technical indicators, considering time periods of 7, 14, and 28 days as predictors. However, to eliminate irrelevant predictors, we employed the random forest variable importance technique. This allowed us to identify the following significant indicators: Relative Strength Index, Commodity Channel Index, Momentum (for time period = 7), William's %R, Ultimate Oscillator, and Rate of Change. These indicators were standardized before being used as input for different models.

The machine learning process consisted of two stages. In the first stage, the models were trained using the technical indicators. In the second stage, the system classified the data based on the trained indicators. The outcome of this analysis was the predicted trend of the market index, which formed the basis for setting trading rules. The following rules were established based on the predicted trends:

- If the next day's trend is an uptrend, the decision is to buy.
- If a buy decision already exists, the decision is to hold.
- If the next day's trend is a downtrend, the decision is to sell.
- If a sell decision already exists, the decision is to hold.

The returns of the strategy were calculated based on the results obtained from these rules.

In our study, we employed a diverse set of machine learning models to predict stock market movements based on the selected technical indicators. The following models, along with an ensemble approach, were utilized in our analysis:

A. K-Nearest Neighbors (KNN):

KNN is a non-parametric classification algorithm that predicts the class of a sample based on the majority class of its k nearest neighbors in the feature space. We trained the KNN model using the standardized technical indicators and evaluated its performance in predicting the buy and sell signals for the stock market.

B.Decision Tree:

Decision trees are hierarchical structures that recursively partition the feature space based on the selected indicators. Each internal node represents a decision based on a feature, while each leaf node represents the predicted class. By training a decision tree model using the technical indicators, we aimed to capture the underlying patterns and rules governing stock market movements.

C.Random Forest:

Random Forest is an ensemble learning method that combines multiple decision trees to make predictions. It constructs a multitude of decision trees and aggregates their predictions to arrive at the final result. By utilizing the random forest model, we aimed to leverage the diversity and robustness offered by the ensemble approach to improve the accuracy and stability of our predictions.

D.Support Vector Machines (SVM):

SVM is a powerful supervised learning algorithm that maps the data into a high-dimensional feature space and identifies a hyperplane that maximally separates the classes. We applied SVM to our dataset, aiming to find an optimal decision boundary that distinguishes between buy and sell signals in the stock market.

E.Gaussian Naïve Bayes:

Gaussian Naïve Bayes is a probabilistic classification algorithm based on Bayes' theorem and assumes that the features are conditionally independent given the class. By training a Gaussian Naïve Bayes model on our dataset, we aimed to estimate the posterior probabilities of different classes, enabling us to make informed decisions about buying or selling stocks.

F. Ensemble Approach:

In addition to individual models, we employed an ensemble approach to combine the predictions from multiple models and generate a more robust and accurate forecast. The ensemble method leveraged the collective wisdom of the constituent models to improve the overall performance. This was achieved by aggregating the predictions from each model, employing techniques such as majority voting or weighted averaging to arrive at the final prediction.

By utilizing this diverse set of models and an ensemble approach, we aimed to capture different aspects of the data and exploit the strengths of each model to enhance the accuracy and reliability of our predictions. The ensemble technique further helped in reducing potential biases and errors associated with individual models, leading to more robust and consistent outcomes.

It is important to note that the models and the ensemble approach were carefully implemented and fine-tuned to achieve optimal performance based on the specific dataset and objectives of our study. The results obtained from these models provided valuable insights into the effectiveness of the momentum strategy and its ability to predict stock market trends. However, further analysis and validation are necessary to assess the generalizability and robustness of the models across different market conditions and time periods.

For the Second Model, we have used Oanda as our currency trading platform. Oanda is an online trading platform that provides access to a wide range of financial instruments, including forex (foreign exchange), commodities, indices, and cryptocurrencies. It offers a user-friendly interface with customizable charts, real-time market data, and advanced trading tools. Oanda is known for its reliable execution, competitive spreads, and flexible trading options, catering to both beginner and experienced traders. Additionally, it provides educational resources and customer support to assist traders in their trading journey. All the historical and streaming data are also being extracted from Oanda for the currency trading section of the model.
Over the many years, trading has been based on several strategies, where few of them are based on Simple Moving Averages, Momentum and Mean Reversion, which also form an important part of our model.

A. Simple Moving Averages (SMA)

Trading strategies based on Simple Moving Averages (SMA) are popular among technical analysts and traders. A simple moving average is a calculation that helps identify the average price of an asset over a specific period of time. By using different time periods for calculating SMAs, traders can analyze price trends and make decisions based on the crossovers and interactions between these moving averages. Two common trading strategies based on Simple Moving Averages include SMA crossover strategy and SMA support and resistance strategy.
Our model uses the SMA crossover strategy where a bullish trend is indicated when the shorter-term SMA (e.g., 50-day) crosses above the longer-term SMA (e.g., 200-day), which, as a result, generates a buy signal. Conversely, when the shorter-term SMA crosses below the longer-term SMA, it generates a sell signal. This indicates a bearish trend and suggests selling or shorting the asset. It's important to note that no trading strategy guarantees profits, and it's crucial to perform thorough analysis, consider risk management techniques, and adapt the strategy to market conditions. Additionally, using multiple indicators or combining SMA strategies with other analysis tools can help improve decision-making.
Our model works with the data for the EUR/USD exchange rate where SMA parameters are optimized for maximum gross performance.

B. Momentum

Momentum strategies can be broadly categorized into two types: cross-sectional momentum and time series momentum, both of which aim to capitalize on the persistence of price trends. However, cross-sectional momentum focuses on relative strength among assets within a universe, while time series momentum focuses on the trend of individual assets over time.

The cross-sectional momentum strategy focuses on comparing the performance of different assets within a specified group or universe. It aims to capture the relative strength of assets within a defined universe and assumes that assets that have been performing well will continue to outperform their peers in the short to medium term.

Time series momentum, also known as trend following, focuses on the price trend of an individual asset over time. It aims to capture and profit from trends in individual asset prices. It assumes that assets that have exhibited strong price momentum in the recent past are likely to continue their trend in the near future.

The most basic time series momentum strategy is to buy if the previous return was positive and sell if it was negative.

C. Mean Reversion

Trading strategies based on mean reversion aim to take advantage of price movements that deviate from their average or mean value and anticipate a return to the mean. The underlying principle is that extreme price movements are often followed by a period of reversal or correction.

Bollinger Bands Strategy and Mean Reversion Oscillator Strategy are two common mean reversion trading strategies. The Bollinger Bands strategy assumes that price deviations from the mean are temporary and that prices will eventually revert to the average. It provides potential trading opportunities when the price reaches extreme levels. The mean reversion oscillator strategy assumes that price movements that have pushed the oscillator into extreme levels are likely to reverse. It aims to capture profits from the anticipated correction back to the mean.

Backtesting a mean reversion strategy in the EUR/USD exchange rate is similar to backtesting SMA and momentum-based strategies. The idea is to set a threshold for the distance between the current stock price and the SMA, which indicates whether to go long or short.

Moving forward, the question of how much capital to deploy to a given algorithmic trading strategy given total available capital is central to algorithmic trading. The answer to this question is dependent on the primary goal of algorithmic trading. Most individuals and financial institutions will agree that long-term wealth maximization is a good candidate objective.

Simply put, the Kelly criterion allows a trader to calculate the fraction of available capital that should be allocated to a strategy based on its statistical return characteristics.

To backtest an algorithmic trading strategy for the EUR/USD currency pair, we propose using an ML-based approach for predicting the direction of market price movements with historical data from the Oanda v20 RESTful API. It employs vectorized backtesting, this time accounting for the bid-ask spread as proportional transaction costs. The backtest is based on intraday data, specifically bars lasting 10 minutes. The ML-based strategy employs several time series features, including the log return and the minimum and maximum closing price. Furthermore, the feature data is delayed. In other words, the ML algorithm must learn from historical patterns represented by lagged features data.

Additionally, sentiment analysis can be used in high-frequency algorithmic trading to gain insights into market sentiment and make faster, data-driven trading decisions. We have incorporated the analysis of news articles and other sources to gauge the overall sentiment surrounding specific assets or markets. Our proposed model extracts the data for sentiment analysis from Refinitiv Eikon. Refinitiv Eikon is a comprehensive financial information and trading platform that provides real-time market data, news, analytics, and trading tools to financial professionals. It offers a customizable interface and advanced features to facilitate data analysis, decision-making, and execution of trades across variousasset classes.

By analyzing sentiment indicators such as positive, negative, or neutral sentiment, algorithms can identify emerging trends, news-driven events, or sentiment shifts that may impact prices. Sentiment analysis can generate sentiment-based indicators that complement traditional technical indicators. These sentiment indicators can be incorporated into high-frequency trading algorithms to generate buy or sell signals. For instance, an algorithm may use a sentiment indicator as a confirmation signal to execute trades when other technical indicators align.

In terms of risk analysis, we propose calculating the risk based on the maximum drawdown and the longest drawdown period. The maximum drawdown is the largest loss (dip) following a recent high. As a result, the longest drawdown period is the time required for the trading strategy to return to a recent high. Value-at-risk (VaR) is another important risk metric. It is expressed in currency and represents the maximum loss that can be expected given a given time horizon and confidence level.

V. RESULTS

A. Result on the historical stock data

We observe that our returns are indeed positive and in line with actual market returns, even though actual returns are slightly greater at times. Among the different machine learning models utilized in our study, the random forest model emerged as the most effective approach for

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 07 Issue: 11 | November - 2023 SJIF Rating: 8.176 ISSN: 2582-3930

predicting stock market movements based on the selected technical indicators.The random forest model demonstrated superior performance in terms of accuracy and stability compared to other models, making it the preferred choice for our analysis.Our model achieved an high accuracy of 96%.

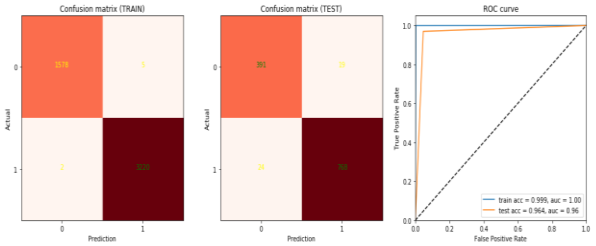

Fig.1 Confusion Matrix and ROC curve

The random forest algorithm constructs an ensemble of decision trees by utilizing bootstrapped samples from the training data and selecting random subsets of features at each split. By aggregating the predictions from multiple trees, the random forest model leverages the wisdom of the crowd and mitigates overfitting issues associated with individual decision trees.

In our experiments, the random forest model achieved a high level of accuracy, surpassing the other models evaluated in our study. The ensemble nature of the random forest allowed it to capture diverse patterns and relationships present in the data, resulting in robust predictions for buy and sell signals in the stock market. Furthermore, the random forest model exhibited good generalization performance, indicating its ability to adapt well to unseen data. By effectively managing the trade-off between bias and variance, the random forest approach minimized both underfitting and overfitting, leading to reliable predictions across different market conditions.

Additionally, the random forest variable importance technique played a crucial role in feature selection by identifying the most relevant technical indicators for predicting stock market movements. This process ensured that the selected indicators contributed significantly to the predictive power of the model, enhancing its overall performance. The returns obtained from the strategy based on the random forest model were positive and aligned with the actual market returns, demonstrating its ability to accurately predict the direction of stock prices. While the actual returns may have been slightly higher in certain periods, the strategy implemented with the random forest model still yielded profitable results.

The high accuracy and favorable returns obtained from the random forest model highlight its effectiveness in capturing the underlying patterns and trends in the stock market. The ensemble nature, feature selection capability, and generalization performance of the random forest model make it a valuable tool for implementing momentum strategies and aiding investment decisions. However, it is important to note that despite the success of the random forest approach, further analysis and evaluation are necessary to ensure its reliability and robustness in different market scenarios and time periods. Additionally, incorporating other evaluation metrics, such as precision, recall, and F1 score, would provide a more comprehensive assessment of the model's performance.

Overall, the results obtained with the random forest model validate its suitability for predicting stock market movements and reinforce its position as the preferred approach in our study.

.

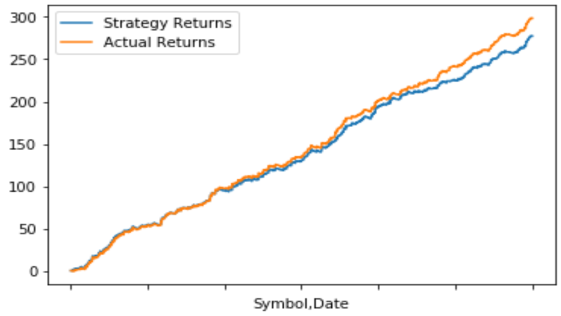

Fig.2 Actual Returns vs Strategy returns

B. Trade initiated on the Oanda platform

Post the implementation of the trading strategy on EUR/USD market deployed by our proposed model, setting up a paper account with Oanda and setting up an appropriate leverage ratio for our account according to the Kelly criterion, we implemented the python code for automated trading. Further, Backtesting begins with retrieving the data. The data set in our executed trade itself is for a single day and has a one-minute granularity. Here, granularity refers to the level of detail or resolution at which data or information is analyzed or represented. It determines the size or frequency of the data points or intervals used in trading analysis. For each minute, the values Open (o), High (h), Low (l), and Close (OHCL) are returned, which form the basis of a common one-minute bar or candlestick structure, along with Tick Volume (volume) and Complete (complete). The output at the end of the relevant code block provides a detailed overview of the data set, which is then used to implement the trading strategy's backtesting.

As a result of backtesting followed by the resultant automated trading strategy, we observed a trade initiated by the python script on the Oanda platform. We used Oanda's trading tools to monitor the executed trades and open positions while the trading algorithm was running. The browser-based trade application is depicted in the following screenshot.

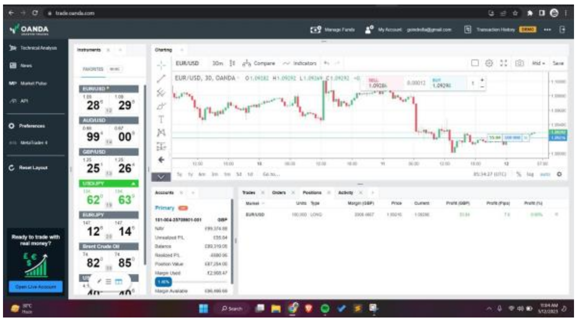

Fig.3 Snapshot of Trade executed.

As an observation, we can say that in consequence to the automated trading order placed, our model resulted in a profit of 55.84 GBP.

VI. CONCLUSION

In a nutshell, our model proposes the automated deployment of an algorithmic trading strategy based on a machine learning classification algorithm to predict the direction of market movements. It covers critical points for consideration such as capital management (based on the Kelly criterion), sentiment analysis, vectorized backtesting for performance and risk, deployment infrastructure, and logging and monitoring during deployment. However, it is important to note that High-frequency trading (HFT) is a complex and highly competitive field that continuously evolves with technological advancements and market conditions. Success in HFT depends on a combination of cutting-edge technology, robust strategies, access to market data, and effective risk management practices.

VII. REFERENCES

[1] I. Parmar, Navanshu Agarwal, Sheirsh Saxena, Ridam Arora, Shikhin Gupta, Himanshu Dhiman, Lokesh Chouhan, "Stock Market Prediction Using Machine Learning," in the *First International Conference On Secure Cyber Computing and Communication (ICSCCC)* ,2018

[2] Leung, C.K.S., MacKinnon, R.K. and Wang, Y., "A machine learning approach for stock price prediction," in *Proceedings of the 18th International Database Engineering & Applications Symposium*, 2021, pp. 274-277. ACM.

[3] Madge, S. and Bhatt, S., 2015., "Predicting Stock Price Direction using Support Vector Machines". in *Independent Work Report Spring*., 2015, pp. 1-4.

[4] Teixeira, L.A. and De Oliveira, A.L.I., 2010. "A method for automatic stock trading combining technical analysis and nearest neighbor classification" in *Expert systems with applications*, 37(10), pp.6885-6890.

[5] G. Amoudi, H. S. Alghamdi, J. Nasser, K. Saeedi and S. Elhag, "Deep Learning Approaches for Detecting Stock Prices: A Survey," in *IEEE Access*, vol. 9, pp. 20235-20254, 2021.

[6] Neelakshi Gupta and Tejinder Kaur, "A new algorithm for classification of stocks based on volality", *International Journal of Advances in Electronics and Computer Science*, vol. 2, no. 9, Sept. 2015.

[7] Bradley Spieler, Catherine Batte, David L. Smith , John-Paul Grenier, "A Automated AlgoTrading Pattern using python," in, *Springer* 2020.