

Stock Market Analysis: Algorithmic Trading

Shivam Sharma¹, Shrey Kaushik², Shubham Srivastava³, Vikrant Jatain⁴, Nagarathna N⁵

^{1,2,3,4}Student, Department of Computer Science and Engineering, B.M.S College of Engineering, Bangalore-India

⁵Assistant Professor, Department of Computer Science and Engineering, B.M.S College of Engineering, Bangalore-India

Abstract - Stock Market Analysis is a platform that is used in the evaluation National Stock Exchange of India with over 5000 Indian companies. Algorithmic trading gives us greater profits with a hybrid model on LS-SVM (Least square support vector machine) to avoid overfitting and combining various modules like analysis, prediction, automated trading, stock guru analysis, search, developer, trend analysis, financial reports, etc.

Particle swarm optimization (PSO) and least square support vector machine (LS-SVM), the LS-SVM is used to predict the daily stock prices along with PSO to optimize it, PSO algorithm helps in selecting the best free parameters combination for LS-SVM to avoid overfitting, local minima problems and improve accuracy in the prediction. The equity sector market returns by algorithmic trading is designed such that it is greater than the bank savings deposit of around 7 percent annually in India.

Key Words: Algorithmic trading, automated trading, Particle swarm optimization,

1. INTRODUCTION

Stock Market is a place where public listed company's shares are traded. In the primary market companies float shares to the general public in an initial public offering (IPO) to raise capital needed by the company to expand or divest their company into new ventures. In India, the secondary and primary markets are governed by the Security and Exchange Board of India (SEBI) which was established on 1988 and given statutory powers in 1992.

The National Stock Exchange, India has a total market capitalization of more than US\$2.27 trillion, the world's 11th-largest stock exchange as of April 2018. NSE's flagship index, the NIFTY 50, is used extensively by investors in India and around the world as a barometer of the Indian capital markets.

The U.S. stock market is currently \$34 trillion, compared to the rest of the world's \$44 trillion capitalization. The U.S. is 43% of world market value, but it houses only 17% of the world's stocks. This huge difference between the Indian stock market and US stock market is an economic indicator. The more the people invest in a country, the better the economy of that country gets. This also is a reason for a developing nation like India to get along with the developed countries.

Algorithmic trading drives about 30% of the trading volumes in Indian equity markets and the percentage is on the rise every day. In the west this percentage is somewhere around 70-80 percent. This gap combined with the low investments from the Indian households is a serious indicator in our economy.

Hence the shift to algorithmic trading with the utilization of Artificial Neural Network for price prediction helps us in dealing with optimizing large sets with multiple parameters. A stochastic model is used like that of Markov model where randomness is a key factor in consideration while making predictions.

The use of machine learning algorithms or machine learning model are proving to be quite useful for stock price prediction, trend analysis, charting, technical analysis while dealing with real time data

2. RELATED WORK

There are several existing systems like quantitative trading or high frequency trading or arbitrage, that are available to large multi corporations, investment banks and hedge fund managers. The systems have been personally tailored to meet the investors requirements and all the fundamental and technical analysis done by the high computing servers which is available to only a selected few people[11]. Whereas the larger number of retail or common investors with a small portfolio, like a common Indian household investor with little capital must rely on traditional means of accessing stock market data with limited analysis to be performed[7].

The lack of mathematical and technical knowledge of a common man also is an inhibiting factor to algorithmic trading or stock market analysis[1]. The existing third party stockbroker's websites, google finance or yahoo finance provide various technical charts and live stock data with no access to perform user algorithms or analysis on the given data[13]. There is just a one-way communication from the websites to users. The ability to perform statistical analysis on the database isn't provided[2]. Access to a company's financial reports must be manual either by visiting the company website or viewing the company profile in the website.[34] Newspapers, financial journals and online websites like economic times provide recommendations from various

stock experts or analysts and other trend details in the current market[9].

Identifying social emotions and detecting the trend through the web data. News & blogs also give us a great insight on current trends[13]. The semantics and affective resources for opinion mining becomes crucial and the structure of data and the way user classifies it[15].

When user obtains such varied forms of data from different sources, we start looking for new financial stock market forecast measures using current data mining technique[8]. Simple LSTM neural network with character level embeddings for stock market forecasting, using only financial news as predictor[29]. We have also come across the usage of optimal Long Short-Term Memory (O- LSTM) deep learning and adaptive Stock Technical Indicators (STIs)[8]. They've also evaluated the model for taking buy sell decisions at the end of the day at around 3 30pm IST for Indian Markets[10].

The several advantages with these implementations are that algorithms like PSO algorithm helps in selecting the best free parameters combination for LS-SVM (Least square support vector machine) to avoid overfitting[18], also they help in selecting the best free parameters combination for LS-SVM (Least square support vector machine) to avoid overfitting[18]. The use of machine learning algorithms or machine learning models are proving to be quite useful for stock price prediction, trend analysis, charting, technical analysis while dealing with real time data[20]. Also the current financial report formats obtained for every company are standardized and easy to extract, compute and analyze[23].

Various disadvantages are that there is no information on the success ratio or the trust factor on the analysts that provide such recommendations or predictions[25] and the information provided in a one-way communication and the user cannot statistically operate on the data[28]. The main problem faced is the uncertain trend in the stock market, the uncertain trends justify that there might be risk involved with stock for various time periods[4]. 4. The twitter sentiment analysis of majority of tweets often is an outcome of the media presence of the company and sentiments don't factor based on company portfolio [32].

3. PROPOSED FRAMEWORK

In the framework, we use a web data scraper with python to fetch, analyse and classify the datasets using algorithms which increases the possibilities of profits. The raw data from various resources are stored in a MySQL Database to be later accessed by the Django Framework based website with templates to give users a great visual experience and also provide necessary tools. The stock data present in the database are also to be organized based on gainers and losers in various periods of time like daily, weekly,

monthly, quarterly, etc. This way we can perform simple analysis that can be easily understood even by a beginner.

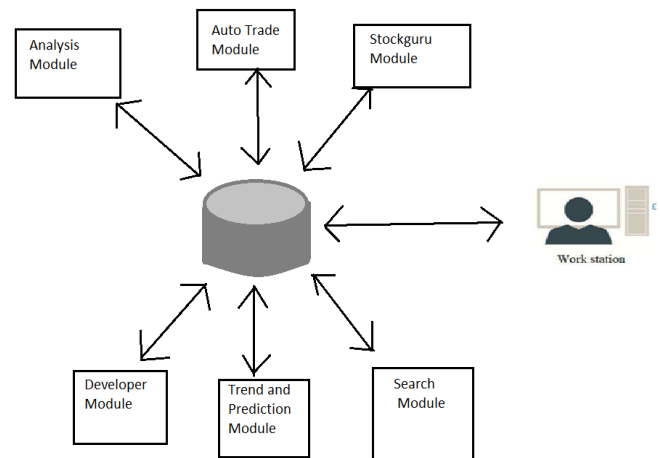


Fig 1 : High Level Module Framework

The implementation is divided into modules. The developer module is for the traders with a little technical SQL knowledge to perform their own statistical analysis on the database of stock data. A search module to get all the stock data with financial reports and technical charts presented in an interactive dynamic web page. The analysis module is to provide simple analysis on the tables present in the database. Prediction module is to combine all the raw data we have obtained and predict the probability of bullish or bearish movement of any stock. Stock guru analysis to determine the success ratio of the analysts who provide recommendations or predictions. Trend analysis and automated buy and sell modules would help users place orders in the same platform with a link to his or her bank and demat account. The integration of all these modules in our python based framework will help us provide a great platform for an investor.

4. IMPLEMENTATION

The project has been implemented in python based Django Framework which consists of a model view template. The model consists of our processed raw data in the form of a structured SQL database, template deals with the UI with html files, view is the link in between the model and template where we send the required data to the template for the user to interact with.

A local Xampp server is used to organise our database to store raw data. This project uses BeautifulSoup, Selenium with Drivers, etc to automate the live extraction and live trading. It takes three hours to fetch all historical data of stock market 5275 companies from Year 2000 to present.

A total of six modules are created and its implementation in detail is mentioned below.

1. Developer Module.

2. Search Module.
3. Analysis Module.
4. Prediction Module.
5. Stock Guru Module.
6. Automated Trading Module.

In the project a tables page has also been implemented to create a simple UI and UX for users to view and interact with the backend database as a two way communication in the project. We are trying to automate trading, give normal users a one stop solution and financial reports need not be fetched manually from company websites.

DEVELOPER MODULE

The developers who have knowledge of sql or even a basic programmer can use this. He/She will be able to execute his own queries on our database to obtain desired results. On google when we search for some company we only get the data, chart, etc. Here we get the raw data and also enable user to communicate with data and process it so that the user can also apply the algorithms.

The user has been provided with input text to insert the sql query in string format. The string is then extracted on submit via POST method to strip it and remove unwanted characters like ‘;’ if present. The query is then made to run on the database present with the help of mysql.connector. We connect to the database and with the cursor run the query that the user has entered.

The results are fetched using cursor.fetchall() which will return records of the entered analysis query if it is present. In case the query entered by user is wrong or doesn't return any records, the corner error scenarios are handled and a placeholder message is sent to convey the error message to the user.

Example: -

Developer runs query “select * from d1g” or “select * from d1g where Current>200;”

The code strips the string and removes unwanted characters. Here d1g represents day 1 gainers, and the query runs on the database to provide users the results of the query. Complex queries are also handled by our module.

SEARCH MODULE

Search module is a one place platform to view financial statements and stock historical data all in one place. The user is supposed to enter the name of the company to search and find some details and analyse them. A “POST”

request is sent and we then match the entered string as a substring of all the company names present in our database, using the query “Select Cticker from Companies where Cname LIKE ‘%{companyName}%’;” This query with string pattern matching returns the ticker for the company that is used to identify the specific company. The Ticker is then used to extract stock records and present it to the user in descending order of the date using “Select * from Cticker ORDER BY Date DESC”. The result given has High, Low, Open, Close, Adjusted Close prices for the mentioned Date and also the volume of shares bought & sold. In case multiple companies are present or if incorrect names are entered by the user, then the error handling cases will return placeholder message to the user.

This module also fetches live data of its financial yahoo finance website. As soon as user enters the company and then we find ticker we replace the ticker in the link : <https://in.finance.yahoo.com/quote/RELIANCE.NS/financials>

The ticker here is “RELIANCE.NS” the python drivers will run selenium code to open the mentioned link and fetch the last three years financial record statements from the website

Income statement <small>All numbers in thousands</small>				
Breakdown	TTM	30/3/2019	30/3/2018	30/3/2017
Total revenue	5,97,20,60,000	5,66,97,60,000	3,91,77,20,000	3,05,14,20,000
Cost of revenue	4,55,84,00,000	4,41,78,60,000	3,07,38,40,000	2,43,57,40,000
Gross profit	1,41,36,60,000	1,25,19,00,000	84,38,80,000	61,56,80,000
Operating expenses				
Selling general and administrative	-	19,59,20,000	14,17,10,000	14,67,20,000
Total operating expenses	74,52,40,000	60,88,30,000	36,75,20,000	26,56,70,000
Operating income or loss	66,84,20,000	64,30,70,000	47,63,60,000	35,00,10,000
Interest expense	19,60,90,000	15,24,70,000	7,24,60,000	3,15,70,000
Income before tax	58,24,10,000	55,22,70,000	49,42,60,000	40,03,40,000
Income tax expense	14,48,00,000	15,39,00,000	13,34,60,000	10,20,10,000
Income from continuing operations	43,76,10,000	39,83,70,000	36,08,00,000	29,83,30,000
Net income	43,36,80,000	39,58,80,000	36,07,50,000	29,90,10,000
Net income available to common s...	43,36,80,000	39,58,80,000	36,07,50,000	29,90,10,000
Basic EPS	-	66.82	60.94	50.66
Diluted EPS	-	66.80	60.89	50.57
Basic average shares	-	59,24,799	59,19,812	59,01,629

Fig 2 : Yahoo financial statements for Reliance Infrastructure Limited – ‘RELIANCE.NS’

ANALYSIS MODULE

This is our analysis module where we now perform preliminary analysis process on our database to generate significant potential companies for users to invest in.

The entire structure of historic stock data have been divided in the time frames as mentioned below

tablelist=["d1g","d2g","d3g","d1l","d2l","d3l","w1g","w2g","w1l","w2l","m1g","m1l","q1g","q2g","q3g","q4g","q1l","q2l"]

```
l","q3l","q4l","hy1g","hy2g","hy1l","hy2l","y1g","y2g","y1l",
"y2l"]
```

The above python code line is to create tables and populate the them in the structure. Each table gives significant values on time and gainers or losers. The time frames are daily, weekly, monthly, quarterly, half yearly, yearly. The numbers mention the timeline in past. The end notation 'g' or 'l' is used to represent gainers or losers.

For example w1g & w2g - It represents the stocks that are weekly gainers in the week 1, current date-5days, and weekly gainers 2 week before- current date-5 to current date -10. All the data are for the trading days and each trading week is generally having 5 trading days.

The list of all stock tickers are obtainer in an excel format. The excel format is worked on using pandas library and we run the api to fetch yahoo financial records of each and every company. Try and except statements are present wherever error scenarios could arise. The records are received in descending order from the current date. The following checks are made and based on

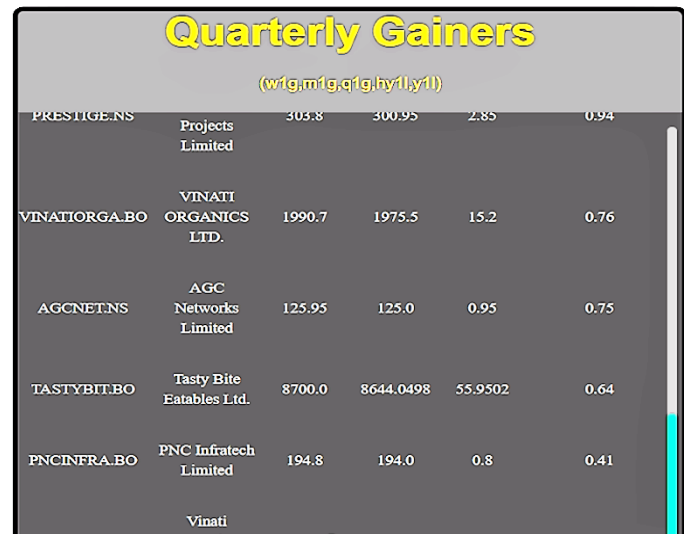
```
currentValue = df['Close'].iloc[total-1]
```

```
previousValue = df['Close'].iloc[total-6]
```

Current and previous closing price. We insert the companies into the mentioned tables. Every company runs its code for all the time frames mentioned and sorts it to place it in those tables.

The analysis module also has implementations to fetch only buyers and only sellers data live from the website, so further analysis can be done. Which companies had only buyers of shares or only sellers with volume data. These are highly demanded (only buyers) and highly avoidable (only sellers) companies.

The assortment is done and now the analysis is done on these tables. Each table mentions its title and its code word.



Quarterly Gainers					
(w1g,m1g,q1g,hy1l,y1l)					
PRESTIGE.NS	Projects Limited	303.8	300.95	2.85	0.94
VINATI.ORG.A.BO	VINATI ORGANICS LTD.	1990.7	1975.5	15.2	0.76
AGC.NET.NS	AGC Networks Limited	125.95	125.0	0.95	0.75
TASTYBITE.BO	Tasty Bite Eatables Ltd.	8700.0	8644.0498	55.9502	0.64
PNCINFRA.BO	PNC Infratech Limited	194.8	194.0	0.8	0.41

Fig 3 & 4 : Recommendations from Analysis Module.

Some of the code words are -

1. All time gainers: (w1g,m1g,q1g,hy1g,y1g)
2. Quarterly gainers: (w1g,m1g,q1g,hy1l,y1l)

For ex: all time gainers with its code word represents all companies which are present in the weekly 1 gainer, monthly 1 gainer, quarterly 1 gainer but also present in half yearly 1 loser and yearly 1 loser. Where 1 represent the immediate previous time frame.

In the similar fashion many other code words are present in the module which are created using complex queries which fetch the records based on the analysis.

STOCK GURU MODULE

A lot of stock guru or analysts come up with predictions like buy shares of a company or sell it. Usually such analysts don't have any success ratios giving us confidence about them. So from economic times we extract all the recommendations on the particular day and analyse their statement by fetching.

Example : "Buy Maruti Suzuki, target price Rs 6,230: Emkay Global"

The string information is used to extract info like Which company they recommend, Who is recommending, When, Whether to buy or sell. Only Buy and Sell are considered in our implementation.

A selenium web browser opens up economic times website : <https://economictimes.indiatimes.com/markets/stocks/r ecos>

We then start obtaining the block of each recommendation and extract our data present in the block with string processing functions in python and applying our logic to segregate recommendations. The data once extracted is again processed one by one to figure on the day of prediction what price it was and how many days it took to

the simulation is done. The final fund is checked for the percentage gain and loss. All such gains and loss are collectively calculated for either a set of top 100 companies or all companies.

The python code uses talib library to get all the parameters we need like RSI, Bollinger Bands, SMA, etc

All time Gainers					
(w1g,m1g,q1g,hy1g,y1g)					
Clicker	Cname	Current	Previous	Change	Change%
ADSL.BO	Allied Digital Services Ltd.	23.57	16.58	6.99	21.04
ADSL.NS	Allied Digital Services Ltd	23.65	16.8	6.85	20.95
AAIL.BO	Aauthum Investment & Infrastruc	66.85	50.0	16.85	23.70
ANKITMETAL.NS	Ankit Metal & Power Limited	1.1	0.85	0.25	29.41
KATRSPG.BO	Katara Spinning Mills Ltd.	18.96	14.87	4.09	27.50
SWADPOL.BO	Swadeshi Polytext Ltd.	51.65	40.6	11.05	27.22
ATNINTER.NS	ATN International Ltd.	0.3	0.25	0.05	20.00

achieve the target. The least number of days it took to reach the target price for buy or sell is searched for in the table and calculated. Companies that don't achieve target have 0 as Achieved.

Some recommendations of companies fetched might not be present in our Database so we filter it out by running a query "select * from stockguru where Achieved IS NOT NULL"

TRADE MODULE

Automated trade execution platform is present where user can enter company name and stock quantity. It is used to execute our trade on HDFC Securities where we have a trading account to place our orders.

Once the user enters the data, the automated selenium driver will open the website and enter the security credentials which are stored in a secret file within the system. The selenium web driver executes the script which then has a sleep time of 2 seconds and then executes the trade of buy or sell.

PREDICTION MODULE

The main prediction module is run by simulating our trade or backtesting it with training data and test data. The backtesting gives us the profitability of the strategy that we use with a graph. The simulation is run such that orders of buy or sell placed on the particular day is printed as an output and the calculation of the initial fund that we started at (Here we put 1000000 Rupees) and then after

BBANDS	Bollinger Bands
DEMA	Double Exponential Moving Average
EMA	Exponential Moving Average
HT_TRENDLINE	Hilbert Transform - Instantaneous Trendline
KAMA	Kaufman Adaptive Moving Average
MA	Moving average
MAMA	MESA Adaptive Moving Average
MAVP	Moving average with variable period
MIDPOINT	MidPoint over period
MIDPRICE	Midpoint Price over period
SAR	Parabolic SAR
SAREXT	Parabolic SAR - Extended
SMA	Simple Moving Average
T3	Triple Exponential Moving Average (T3)
TEMA	Triple Exponential Moving Average
TRIMA	Triangular Moving Average
WMA	Weighted Moving Average

Fig 5 : List of technical indicators

The code snippet to obtain the indicator values are:-

```
close = price['Adj Close'].values
up, mid, low = talib.BBANDS(close, timeperiod=20, nbdevup=2, nbdevdn=2, matype=0)
rsi = talib.RSI(close, timeperiod=14)
sma = talib.SMA(close)
```

Various such indicators are combined in a hybrid model to run the simulation on the stock data for a single company to calculate the difference of final fund and initial fund. The loss making companies are put in one set and the profitable ones. The profits and losses of each company are considered to measure overall profit or loss and gain insights on our strategy.

5. RESULTS

The implementations of all these modules which are integrated together gives us a one stop platform for all the

needs of the investor. We have provided a two-way communication platform where users can interact with the system and directly query the database to obtain the results. This platform has also increased the transparency into the stock analysts by calculating the prediction success ratio. Preliminary analysis gives a user an easy explanation of the trend in the stock market with the appropriate code words. The user can also view the entire financial records of the companies for the past 3 years and also go through the historic stock prices of about 5275 companies in the Indian stock market.

The user also has access to the pre-processed database with all the gainers and losers of all time frames from daily, weekly, monthly, quarterly, half yearly and yearly.

The model created by combining various parameters as mentioned in this strategy when run on a range of companies gives us an estimate of profit to loss ratio. The profitability which is measured by the difference in the final funds to the initial funds gives us data representing about 2/3rd of profitable companies and 1/3rd of loss making companies. When cumulative is considered the strategy is manipulated again and applied to specific top 100 companies to give us a higher number of profit making companies and hence the strategy can be applied by the user to trade automatically.

The testing results are :

```
Python 3.7.5 Shell
File Edit Shell Debug Options Window Help
Final Funds 9802.959877929688

= RESTART: C:\Users\shubh\AppData\Local\Programs\Python\Python37\testingvwap.py
Company is bajajfinsv_ns
Bought at 5818.4501953125 5781.9914179208745 4181.5498046875 2018-11-28
0.01345605257995451
Drop 0.9208746758871863 0.031934215349927624 5902.5
Sold at 5902.5 5887.623123629704 10084.0498046875 2018-12-05

Bought at 6152.64990234375 5911.6856312063 3931.39990234375 2018-12-18
0.010214700579444126
Drop 0.5156414946961162 0.07626818544289352 6368.7998046875
Sold at 6368.7998046875 6141.910598953236 10300.19970703125 2019-01-14

Bought at 7279.60009765625 6344.70426222022 3020.599609375 2019-04-02
0.010196362831966085
Drop 0.6219740659641534 0.2353102349973124 7593.0498046875
Sold at 7593.0498046875 6972.796085027477 10613.6494140625 2019-07-08

Bought at 7855.39990234375 7006.25073956378 2758.24951171875 2019-07-09
0.010465986360561428
Drop 0.8720528576331716 0.1211987972375706 7162.4501953125
Sold at 7162.4501953125 7053.077836832204 9920.69970703125 2019-07-22

Final Funds 9920.69970703125

= RESTART: C:\Users\shubh\AppData\Local\Programs\Python\Python37\testingvwap.py
Company is 3MIndia bo
Bought at 20833.900390625 20594.338907320205 -10833.900390625 2018-11-28
0.01142874659297332
Drop 0.9282892479123697 0.07133072734806338 22163.5
Sold at 22163.5 22050.706663969348 11329.599609375 2018-12-05

Bought at 24144.80078125 22453.221197570387 -12815.201171875 2019-04-01
0.012743694488104128
Drop 0.6529962050831248 0.11391078068284619 23477.599609375
Sold at 23477.599609375 22584.87650893246 10662.3984375 2019-04-26

Final Funds 10662.3984375
```

Fig 6 : Trade simulation for 'Bajaj Finserv Ltd' (Loss Rs 80 in 8 trades) and '3M India Ltd' (Profit Rs 662 in 4 trades)

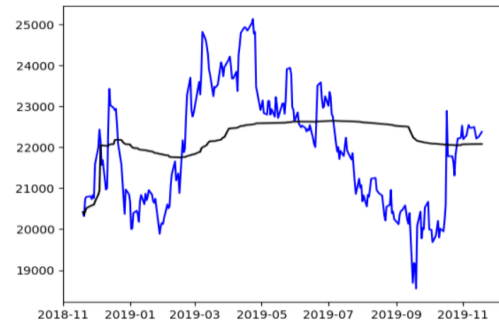


Figure 7: Vwap vs stock price trading
x-axis: days, y-axis: price in Rs,
black line: Vwap, blue line: live stock price.

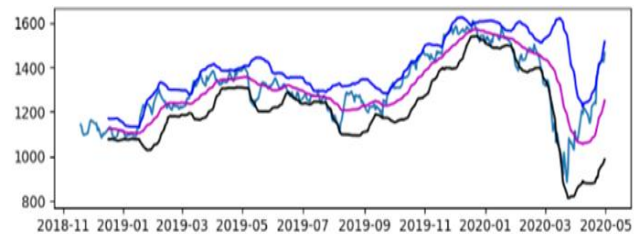


Figure 8 : Bollinger bands vs stock price.

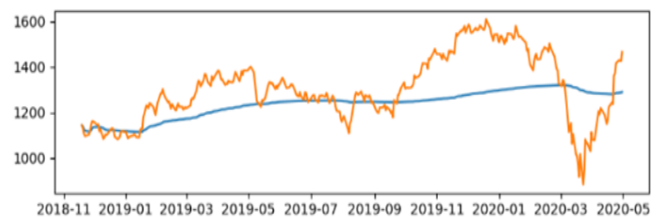


Figure 9: Vwap vs stock price.

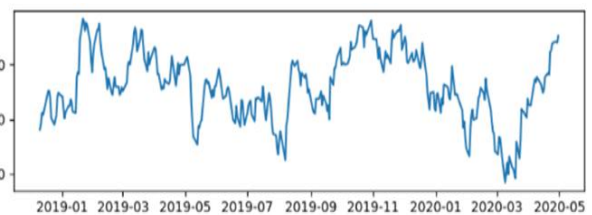


Figure 10: RSI (Relative Strength Index 0-100)

6. CONCLUSIONS

The proposed framework has generated greater cumulative overall profits when traded with multiple companies using the proposed algorithms. The final funds after simulation gives us a profit margin of greater than 7 percent with respect to initial funds of Rs 10,00,000.

The importance of stock market analysis through the use of computational power will give us better insights into the financial world.

Having a user-friendly interface is one of the important key aspects. We are aiming at providing one platform for all the queries, a new user could have while going through the stock market. In this project, our main aim is to provide a one stop solution for the users of the stock market of India or any country in the world. The more the people invest the more the country progresses.

Users can make a handsome amount of return on investing in stocks or trading in the stock market. All they need is some patience and have to delve into and adapt to algorithmic trading so that they can gain some experience and invest smartly.

REFERENCES

- [1] Rajesh V. Argiddi 1 , Dr.Mrs.S.S.Apte2 , Bhagyashri U. Kale3 "An Analysis on Stock Market Intelligence and Research Approaches", 2014 Walchand Institute of Technology, Solapur, India
- [2] Hong Keel Sul, Alan R. Dennis, Lingyao (Ivy) "Trading on Twitter: The Financial Information Content of Emotion in Social Media", 2014 47th Hawaii International Conference on System Science
- [3] Senthamarai Kannan, P. SailapathiSekar, M.MohamedSathik and P. Arumugam, "Financial stock market forecast using data mining Techniques", 2010, Proceedings of the international
- [4] Leung, Carson Kai-Sang, Richard Kyle MacKinnon, and Yang Wang. "A machine learning approach for stock price prediction." Proceedings of the 18th International Database Engineering & Applications Symposium. ACM, 2014.
- [5] E. Cambria, T. Mazzocco, A. Hussain, Application of multi-dimensional scaling and artificial neural networks for biologically inspired opinion mining, *Biol. Inspired Cogn. Architect.* 4 (2013) 41-53.
- [6] F. Bießmann, J.M. Papaioannou, M. Braun, A. Harth, Canonical trends: detecting trend setters in web data, in: *International Conference on Machine Learning*, 2012.
- [7] E. Cambria, C. Havasi, A. Hussain, Senticnet 2: a semantic and affective resource for opinion mining and sentiment analysis, in: *FLAIRS Conference*, 2012, pp. 202-207.
- [8] M. Bautin, L. Vijayarenu, S. Skiena, International sentiment analysis for news and blogs, in: *Proceedings of the International Conference on Weblogs and Social Media*, 2008
- [9] E. Cambria, B. Schuller, Y. Xia, C. Havasi, New avenues in opinion mining and sentiment analysis, *IEEE Intell. Syst.* (2013).
- [10] K. Senthamarai Kannan, P. SailapathiSekar, M.MohamedSathik and P. Arumugam, "Financial stock market forecast using data mining Techniques", 2010, Proceedings of the international multiconference of engineers and computer scientists.
- [11] Leonardo Dos Santos Pinheiro and Mark Dras "Stock Market Prediction with Deep Learning: A Character-based Neural Language Model for Event-based Trading", 2017 Macquarie University Capital Markets CRC
- [12] Manish Agrawal, Asif Ullah Khan, Piyush Kumar Shukla "Stock Price Prediction using Technical Indicators: A Predictive Model using Optimal Deep Learning", 2019 International Journal of Recent Technology and Engineering (IJRTE)
- [13] Vivek Rajput, Sarika Bobde "Stock Market Forecasting Techniques: Literature Survey", 2016 International Journal of Computer Science and Mobile Computing
- [14] Price Prediction of Share Market using Artificial Neural Network (ANN), Zabir Haider Khan Department of CSE, SUST, Sylhet, Bangladesh.
- [15] Stock Market Prediction Using Hidden Markov Model, Poonam Somani Department of Computer Engineering and Information Technology College of Engineering Pune Maharashtra, India.
- [16] Sudheer V. "Trading Through Technical Analysis: An Empirical Study From Indian Stock Market", 2015 International Journal of Development Research
- [17] Alice Zheng, Jack Jin "Using AI to Make Predictions on Stock Market", Stanford University Stanford 2014
- [18] K. Hiba Sadia, Aditya Sharma, Adarrsh Paul, Sarmistha Padhi, Saurav Sanyal "Stock Market Prediction Using Machine Learning Algorithms", 2019 International Journal of Engineering and Advanced Technology (IJEAT)
- [19] L P Ni, Z W Ni, Y Z Gao. Stock trend prediction based on fractal feature selection and support vector machine[J]. *Expert Systems with Applications*, 2011, 38(5): 5569-5576.
- [20] P W Ling. The stock price forecasting comparative research of the use of fractal theory at Taiwan traditional industry and technology industry[J]. *Applied Mechanics and Materials*, 2013, 274: 53-56.

- [21] Y Yang, G Chen. Stock prices prediction research of BP algorithm[C]//Proceedings of the 2012 International Conference on Cybernetics and Informatics. Springer New York, 2014: 20332040.
- [22] The Trend Analysis of China's Stock Market Based on Fractal Method and BP Neural Network Model WU Bing-hui HE Jian-min School of Economics and Management, Southeast University, Nanjing 211189, P.R.China. B. Chauhan, U. Bidave, A. Gangathade, S. Kale, Stock Market Prediction Using Artificial Neural Networks,
- [23] International Journal of Computer Science and Information Technologies (IJCSIT), 2014, pp. 204-207. Akinwale Adio T, Arogundade O.T and Adekoya Adebayo F, Translated Nigeria stock market price using artificial neural network for effective prediction. Journal of theoretical and Applied Information technology, 2009.
- [24] K. Schierholt and C. Dagli, Stock Market Prediction Using Different Neural Network Classification Architecture, Computational Intelligence for Financial Engineering.
- [25] Osman Hegazy, Omar S. Soliman and Mustafa Abdul Salam "A Machine Learning Model for Stock Market Prediction", 2013 International Journal of Computer Science and Telecommunications
- [26] V Kranthi Sai Reddy "Stock Market Prediction Using Machine Learning", 2018 International Research Journal of Engineering and Technology (IRJET)
- [27] Shen, Shunrong, Haomiao Jiang, and Tongda Zhang. "Stock market forecasting using machine learning algorithms." Department of Electrical Engineering, Stanford University, Stanford, CA (2012): 1-5.
- [28] Bonde, Ganesh, and Rasheed Khaled. "Extracting the best features for predicting stock prices using machine learning." Proceedings on the International Conference on Artificial Intelligence (ICAI). The Steering Committee of The World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp), 2012.
- [29] S. Bao, S. Xu, L. Zhang, R. Yan, Z. Su, D. Han, Y. Yu, Mining social emotions from affective text, IEEE Trans. Knowl. Data Eng. 24 (2012) 1658-1670.
- [30] Hegazy, Osman, Omar S. Soliman, and Mustafa Abdul Salam. "A Machine Learning Model for Stock Market Prediction." arXiv preprint arXiv:1402.7351 (2014).
- [31] Dai, Yuqing, and Yuning Zhang. "Machine Learning in Stock Price Trend Forecasting." (2013).
- [32] Vantage, Alpha., Alpha Vantage - Free APIs For Realtime And Historical Financial Data, Technical Analysis, Charting, And More! Alpha Vantage.co. N.p., 2017. Web. 19 Nov. 2017.
- [33] Dev Shah, Harun Alsah, Farhana Zulkernine "Stock Market Analysis: A Review and Taxonomy of Prediction Techniques", 2019 International Research Journal of Financial Studies
- [34] Abbasi, H. Chen, A. Salem, Sentiment analysis in multiple languages: feature selection for opinion classification in web forums, ACM Trans. Inform. Syst. (TOIS) 26 (2008) 12.