Proceedings of the International Conference on Smart Electronics and Communication (ICOSEC 2020)
IEEE Xplore Part Number: CFP20V90-ART; ISBN: 978-1-7281-5461-9

CUDA parallel computing framework for stock market prediction using K-means clustering

Sangeeta Kumari, Nikhil Patil, Piyush Nankar, Mahesh Kulkarni
sangeeta.kumari@vit.edu
Department of Computer Engineering
Vishwakarma Institute of Technology, Pune-37

Abstract– **With the recent advancement of machine learning algorithms over past two decades, it has gained a lot of attention of academicians and researchers community and also found prominent application in multiple domain including finance. Our proposed model uses Machine Learning algorithms along with Parallel Computing processing to provide a new trading technique for non-stationary and multidimensional financial data of Reliance Industries retrieved from Dematerialized account from upSTOX Application Programming Interface (API) which extract data at regular interval of 10 minutes. This proposed model uses K-means machine learning technique to models the gathered stock data and predicts the upcoming stock values well in advance with parallel processing techniques. This paper provides the analysis of previous year's stock market pricing data and interpret results after performing intensive training based on machine learning algorithm on Compute Unified Device Architecture (CUDA) considering the time constraint of real time trading. The performance of the system is improved drastically with the help of machine learning techniques and to accelerate the process of generating the results, a technique of parallel computing is used in this paper. The performance time is significantly reduced because of high performance speed using CUDA parallel computing technology compared to traditional methods of single Central Processing Unit (CPU). It helped in reduction of calculation time by large margin and hence to gain book profit which is the ultimate goal of trading by predicting the stock values well in advance. Investors can decide whether to keep that stock, sell it or buy some other new stocks or neutral decision on basis of 3 clusters as predicted k means algorithm. Neutral decision means if user owns stock then he should hold that stock with him and if he does not possess stock then he should not buy it. This proposed method is suitable for intraday trading as the stocks value are taken for each 10 minutes on the basis of that model can take decision what should do.**

Keywords: Financial data, stock market, stock prediction, Machine Learning, K-Means Clustering, stock market prediction, Stock forecasting.

I. INTRODUCTION

Many studies suggest that it is quite difficult task to predict the patterns in stock pricing because of its size, dimension, non-stationary and dynamic nature of data. This paper is currently focusing on creating a model based on machine learning algorithms and innovative parallel computing techniques which will support real time trading with accuracy trading.

It has been observed that most of proposed trading models take all the previous stock data present at the moment into consideration as a training dataset. It has the unique advantage that a model gets to deal with all variety of data, but at the same time it reduces the accuracy of prediction [2] [7]. Also this older approach is not very efficient as doesn't help in real time stock market trading because it's inability to produce real-time prediction of stock pricing. To tackle this issue, a revised method is proposed by considering only that data which has resemblance with current stock data rather than using all the available data for training the machine learning model. So for building the model, K Means Clustering algorithm checks for related data which is similar to testing data in a database. But the obstacles don't end here, the requirements of real time stock pricing prediction vary from that of a pre-learned trained model with previous data. Considering the previous statement in detail, to finish the training of a model that feeds on all previous data before the stock market opens is not possible practically, but in case of real time stock market price prediction system it creates the model as soon as the new data is created which is unique also. But it comes with huge computing time to do such a tedious job of complex calculations. It brings more problems in creating and training the model in such a small time frame, which in turn will cost investors with gaining less profit and that is not the motive. So to accelerate the process of performing such complex calculations, the parallel computing techniques such as NVidia CUDA are used to accelerate prediction and will give clear ideas about investments in that particular stock.

The basic idea behind the main algorithm used in this paper is that K means clustering divides the dataset in a number of predefined K values. Each data point belongs to only a single cluster [3]. It focuses on keeping similar data points in the same cluster and separating different clusters as far as possible from other clusters. Main motive is that, the less the variation in data points, the more similar data points in the same cluster becomes. It shadows other traditional clustering algorithms such as density-based and expectation-maximization. It drastically increased model accuracy in experimental results. In order to tackle the problem of consuming huge amounts of time in calculation, it has resulted with the idea of using parallel computing techniques such as NVidia CUDA [8]. This framework helps to build the model in parallel reducing huge computation time.

Rest of the paper is divided as following section: The 2nd section provides insight about literature survey i.e. all the related tasks done in this area. The 3rd and 4th section puts light on the experimental methodology used in this paper and model building of real time trading systems. 5th section takes us through the experimental results obtained in this extensive training model. 6th section talks about the conclusion obtained from this experiment and the end section sums up the references used for this project.

II. RELATED WORKS

Traditional methods of stock market prediction is not appropriate because of non-linearity and non-stationary attributes of data. Our paper is predominantly based K means Clustering and OpenCL parallel Computation. This section will mainly pay attention on similar previous works about different techniques in financial and economic applications.

a) Machine Learning algorithm in financial field

Authors Hx He et. Al. [3] proposed a metrics Trading based on Trend Prediction (TTP) based on K means clustering and regression models for partition of time series financial data. They have kept fixed numbers of clusters, numbers of K means runs and length of time series for analysis of stock trend prediction of United States (US), United Kingdom (UK), Taiwan, Singapore, Canada and Japan.

Authors *Victor Changa, Taiyu Lib and Zhiyang Zengb* [1] used machine learning based Adaptive Boosting algorithm for Prediction of stock market. Adaptive Boosting algorithm is based on adaptive suppression of weak hypothesis using AdaBoost algorithm and thus try to forecast correctly. Authors mainly focused on comparison of profit parameters on for non-stationary data (financial data) using Adaptive Boosting algorithm.

Sun Yutong and Hanqing Zhao [2] have used AdaBoost algorithm to provide a model based on machine learning algorithm considering multi factor for analysis of dynamic and complex data of stock market. They have also compared basic AdaBoost algorithm and advanced AdaBoost algorithm on basis of accuracy and realism and shown that how advance AdaBoost algorithm outperforms basic AdaBoost algorithm. These authors have calculated precision and recall for advance and basic AdaBoost algorithm along with graphical representation of degree of assurance of one fourth top stock.

S.R. Nanda et. al. [7] used K-mean clustering algorithm for portfolio management of Indian stocks of BSE market and also compared the same with Self organizing maps and fuzzy C means. They also compared the returned booked on various portfolios created by k means clustering considering.

Tamal Datta Chaudhuri and Indranil Ghosh [10] proposed a model based on K means clustering along with Gaussian model and self-organizing maps to check volatility of Indian stock mark. They have considered two years of Indian stock data to validate their model by varying number of clusters from 2 to 11 and no. of features from 2 to 9 and then they have represented Dunn Index(DI) value through graphs

With the advancements in machine learning algorithms, numerous works [9] [10] [12] [13] have been carried out adopting these algorithms for financial data over past two decades which includes decision making in stock market, credit rate calculation, fraud detection and Corporate Failure Prediction (CFP). Author Cortés, E. A. et. al. used classification tree on top of layered AdaBoost framework with by Cortes [11], which improved accuracy to significant . Another author Alfaro [12] combined AdaBoost algorithm along with neural network and compared their result with single neural network. This paper has shown thirty percent improved in performance by using neural network with AdaBoost. Authors [13] performed the Financial Distress Prediction (FDP) on data of more than 600 companies of China. They found that result of AdaBoost algorithm is best with decision tree.

Over the past two decades many researchers have used ML algorithms as Regression, Support Vector Machine (SVM), K means cluster, AdaBoost algorithm etc. to improve Performance in financial data but the latency of this algorithm is a major concern. This paper proposes a novel idea that exploits the advantage of AdaBoost algorithm with parallel processing to improve latency for real-time trading system. But the problem here is time consumption, as AdaBoost requires a lot of time on building the model. While in our application speed of prediction is going to be an added advantage because in real time trading, the prediction should be performed rapidly otherwise there is high chance of losing the opportunities of booking profits. So to compensate our research adopts Open Computing Language (OpenCL) parallel computing module. With help of previous research it is clear that parallel computing is being used in many areas for applications such as medical, communication, financial field and computer science.

b) Parallel Computing in financial field

Parallel Computing is mode of computation in which multiple instructions are processed simultaneously in the processor. It is being widely used since the evolution of Big Data. In traditional method of computing a single processor handles the request of process in step by step manner. But nowadays plenty of computer instruction can be handled simultaneously with parallel computing which largely reduces the time required for computation.

Parallel computing is important factor because in real time trading if it cannot provide the proper prediction in particular time, the opportunity of gaining profit will be lost. For that four kinds of framework were created by Thulasiram, Rahman, and Thulasiraman (2003). They made Back Propagation Neural Network (BPNN) algorithm run in parallel with help of multiple threads. It was observed that parallel computing helped getting higher accuracy in less time. Authors Arce, Paola & Maureira-Fredes, Cristián &

Bonvalle[15] implied parallel processing on Feed forward neural networks to perform analysis on dynamic real time, multi factor financial data. In this paper, authors have used ZeroMQ library of CUDA to develop an algorithm using C++ language and this algorithm was tested on NVidia Graphics Processing Unit (GPU) system.

Paper [16] presents a novel idea of detecting states of intraday financial data to determine cluster temporal periods using maximum likelihood approach. They have done case study of temporal behavior of South African equity market to computed State signature vectors using Euclidian distance and shown their result is based on Gephi graph visualization tools.

III. METHODOLOGY

There are many technical Indicators which is the deciding factor in trading of stocks. These indicators help us to find out whether there is a bullish or bearish trend of a particular stock in the market. And once the market trend in known by the values of technical indicators that perform sentiment analysis of different tweets on twitter. And by using both of the results, a final decision has been made at a particular point of the day.

These technical indicators is generally categorized into three categories as **trend indicators, volume and volatility based indexes**.

1. Trend Indicators

Every trend indicator tells us in which direction the market is moving, if there is a trend in the market. And by using those indicators, trends in the market can be easily detected. There are many trend indicators but in this paper's scope consider only the below 4 indicators.

i) Average Directional Index (ADX)
ii) Simple Moving Average (SMA)
iii) Weighted Moving Average (WMA)
iv) Exponential Moving Average (EMA)

a. Average Directional Index

This indicator is used to check strength of trend

$$ADX = \frac{|DX|}{t} \qquad (1)$$

$$DX = \frac{|(+DI(t)) - (-DI(t))|}{|(+DI(t)) + (-DI(t))|} \qquad (2)$$

$$+DI(t) = \frac{+DM(t)}{TR(t)} \cdot 100 \qquad (3)$$

$$-\mathrm{DI}(t) = \frac{-\mathrm{DM}(t)}{\mathrm{TR}(t)} \cdot 100 \qquad (4)$$

This indicator is developed by wilder and according to him ADX value above 25 indicates trend as strong trend and ADX value below 20 indicates that trend is not strong. Here, DM means directional movement and it helps to indicate whether a trend is moving upward or downward.

Where + DM is Plus Directional Movement, - DM is Minus Directional Movement, + DI is Plus Directional Indicator, - DI is Minus Directional Indicator. TR is True Range. DM is the maximum degree of stock price volatility today which are higher than those of the previous day. If the volatility of stock price today is lower than that of previous day, DM equals 0. TR is the maximum stock price volatility based on daily highest price, daily lowest price and daily closing price of the former day.

b. Simple Moving Average:

Simple Moving Average (SMA) is a technical indicator that tells us whether stock price will continue or reverse the bear or bull trend. For calculation of SMA, the closing prices of stock are generally considered. To classify the trend, k-means clustering algorithm is used.

$$SMA_n = \frac{\sum closing\ price\ in\ past\ n\ days}{n} \qquad (5)$$

Where n is number of days. The "n" is considered as any value, but generally it is considered as 14 as default value.

c. Weighted Moving Average

Weighted Moving Average is just the same as SMA but here different weights are assigned to different days in WMA by considering the most recent will affect more on the next value. So, more weight is given to recent day value and less weight to earlier value.

$$WMA_t = \frac{Close_{(t)} \cdot W_{(1)} + Close_{(t-1)} \cdot W_{(2)} + \cdots Close_{(t-n+1)} \cdot W_{(n)}}{W_{(1)} + W_{(2)} + \cdots W_{(n)}} \qquad (6)$$

Where W (t) denotes weight factor, Close $_{(t)}$ is Closing price on that day.

d. Exponential Moving Average(EMA) :

Exponential Moving Average (EMA) also places more weight and significance on recent values like WMA.

$$EMA_{(t1)} = EMA_{(t0)} + \alpha \cdot (p - EMA_{(t0)}) \qquad (7)$$

$$\alpha = \frac{2}{(N+1)} \qquad (8)$$

Where EMA $_{(t0)}$ is Exponential Moving Average of previous time period, α is Exponential smoothing constant, p is Current price. This is an index smooth moving average price over a period of time.

2) Volume Indicators

Volume indicators tell us how volume is changing over time. It is important because when the price changes by seeing volume's value trader can decide how strong the move is. The money flow index is considered as a volume indicator for trend analysis.

a. **Money Flow Index:**

It mainly tells us whether stock is overbought or oversold. MFI value below 20 indicates that stock is getting oversold and MFI value above 80 tells us that stock is getting overbought.

$$\mathbf{MFI} = 100 - \frac{100}{1+MoneyRatio} \quad (9)$$

$$\mathbf{Money\ Ratio} = \frac{P}{N} \quad (10)$$

Where typical price = (daily high price+ daily low price + daily closing price) / 3 and Cash Flow= Typical price * volume. P is the positive cash flow in past N days and N represents negative cash flow in past N days. It also used to spot divergence in price. It uses prices and volume data to generate overbought or oversold signals.

Where, P= Positive Money flow in 14 periods

N= Negative Money flow in 14 periods.

3) Volatility Indicators

Volatility indicators is used to measure the volatility of raw financial time-series sequence data during period of time. The sequence data adopted in this paper includes trading price, trading volume, moving average of the data mentioned, and so on. The large spike in volatility indexes maybe observed when the stock markets is in the rising trend, which also concludes that the market emotion and behavior will likely to overreact.

It tells how much the value changes in a given period. Higher the value of volatility, the price is changing faster. It tells us where prices will go in future. There are several volatility indicators, her most basic indicators like (a) Relative Strength index (b) Average True Range (c) William Overbought /Oversold are considered.

a. Relative Strength Index :

It tells us about overbought and oversold conditions about stock. If RSI value above 70% it indicates overbought condition and when RSI value above 30% indicates oversold condition.

$$RSI = 100 - \left(\frac{100}{1+RS}\right) = \frac{AU_n}{AU_n + AD_n} \cdot 100\% \quad (11)$$

$$RS = \frac{AU_n}{AD_n} \quad (12)$$

$$AU_n = \frac{\sum rising\ points}{n} \quad (13)$$

$$AD_n = \frac{\sum falling\ points}{n} \quad (14)$$

Where, AU_n is Average Upward Price Change, AD_n is Average Downward Price Change

b. **Average True Range (ATR)**

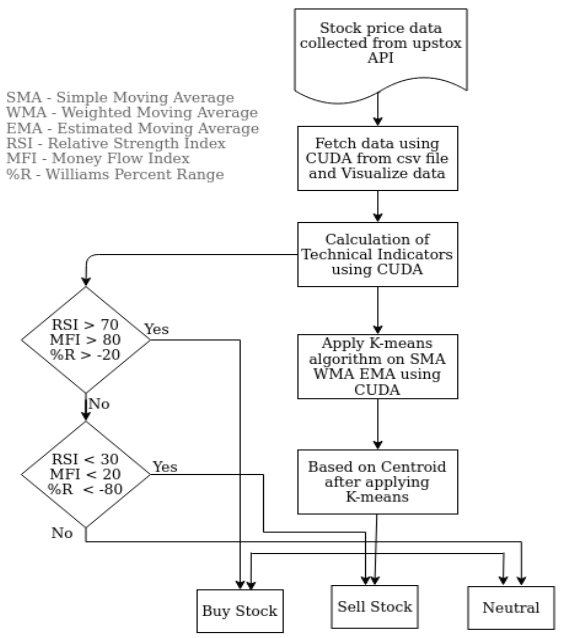
Average of true ranges over the specified period is called the Average True Range.

$$ATR_{(t)} = \frac{TR_{(t)} + TR_{(t-1)} + \cdots TR_{(t-n+1)}}{n} \quad (15)$$

$$TR_{(t)} = MAX\left(ABS(H_t - L_{(t)}), ABS(H_t - C_{(t-1)}), ABS(L_t - C_{(t-1)})\right) \quad (16)$$

IV. Working and Simulation of Proposed Model for Real Time Trading

Figure1: Flowchart for illustration of proposed trading model



a. **Williams Overbought/Oversold Index(%R) :**

Momentum indicator that shows overreacting situation and trading signals. Values of %R moves between 0 to -100.

$$\%R = \frac{(daily\ closing\ price - high\ price\ in\ n\ days)}{(high\ price\ in\ n\ days - low\ price\ in\ n\ days)} \cdot 100 \quad (17)$$

If the value of %R is above -20 it shows overbought and value below -80 shows oversold.

Euclidian distance is used to find distance between centroid and particular value of sma, ema or wma and on the basis these parameters it is assigned to particular cluster.

A. CUDA:

CUDA (Compute Unified Device Architecture) is a model developed by NVidia where it gives access to developers to the GPU's virtual instruction set and to execute the kernel functions. By using CUDA, the speed of computations are increased and reduces the time required for running applications significantly. It also helps to use deep learning and machine learning algorithms efficiently. In this paper, CUDA platform is adopted to accelerate the calculation of technical indicators which relatively increases performance of k-means algorithm. As it is a real time system, small delay may also cost us more so this parallel computation using CUDA can dramatically decrease delay, which can also decrease the risk of losing trade due to delay.

B) Parallel computation on programming model

CUDA has two parallel programming models: Task parallel and Data parallelism. Task parallelism is to execute functions on different cores simultaneously. Data parallelism is like Single Instruction Multiple Data (SIMD). Generally the simple program performs addition linearly using a loop. CUDA program performs addition by using vectors and allocates memory to variables on GPU by using "cudamemcpy", the GPU function is launched by mentioning blocks and threads, and all GPU functions are defined as "__global__". Memory of GPU is called Device memory and memory of CPU is called Host memory.

V. Experimental Results

Here, the main focus is on reducing time to load Open High Low Close (OHLC) values from csv file using CUDA using c and store them in vectors. The technical indicators are calculated using these vectors. To improve system performance of K-means algorithm, simultaneous visualizations of OHLC values are proposed and technical indicators using python which uses the matplotlib library. CUDA parallel computing platform is adopted to accelerate the K-means clustering algorithm because performing computation serially will be more time consuming during its trading period.

A. Data set

In this paper, the data set is retrieved using API from upstox official website. Upstox is a one of the online stock brokers. Data is then saved into a CSV file that includes daily opening, closing, high and low price at the interval of 10 minutes each for 180 days' time frame. Technical indicators are calculated based on the open, low, high and close prices. It has then provided data to the k-means algorithm.

Figure 2. Screenshot of Stock price values retrieved from upstox API at regular interval of 10 minutes

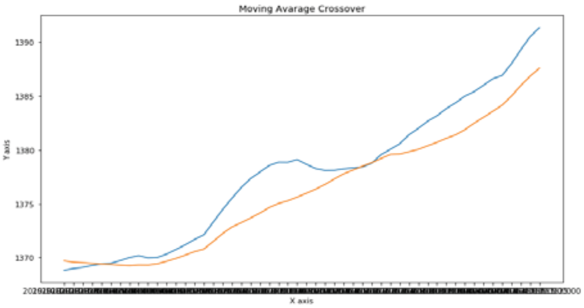
	Timestamp	open	high	low	close
0	2019-10-14 09:15:00	1356.850098	1360.850098	1353.099976	1354.300049
1	2019-10-14 09:25:00	1354.400024	1355.699951	1352.599976	1354.500000
2	2019-10-14 09:35:00	1354.250000	1355.500000	1353.050049	1354.699951
3	2019-10-14 09:45:00	1354.699951	1357.000000	1353.699951	1356.750000
4	2019-10-14 09:55:00	1356.750000	1358.000000	1354.000000	1355.850098
..	...	...	...	...	...
147	2019-10-17 14:45:00	1388.900024	1392.000000	1388.450073	1389.000000
148	2019-10-17 14:55:00	1389.000000	1396.250000	1388.800049	1396.050049
149	2019-10-17 15:05:00	1396.050049	1399.000000	1395.000000	1398.250000
150	2019-10-17 15:15:00	1398.350098	1399.000000	1396.000000	1397.150024
151	2019-10-17 15:25:00	1397.099976	1397.099976	1394.050049	1395.000000

B. Technical Indicators

Parallel computed technical indicators are visualized using python to make decisions.

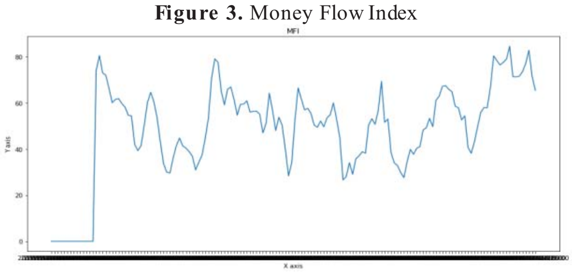
1. Moving average crossover: it is an indication that trend is about to change.

Figure 3. Moving average crossover



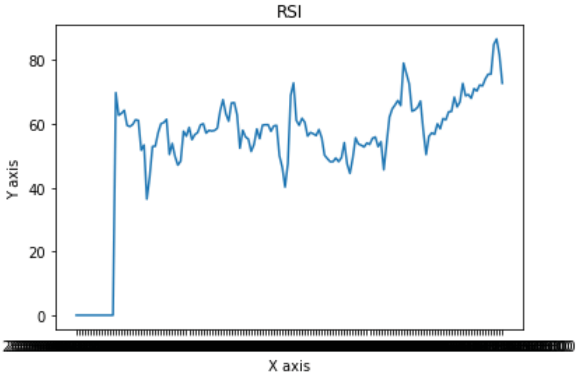
2. Money Flow Index: It mainly tells us whether stock is overbought or oversold and also used to spot divergence in rice.

Figure 3. Money Flow Index



3. Relative Strength Index (RSI): It tells us about overbought and oversold conditions about stock. If RSI value above 70% it indicates overbought condition and when RSI value above 30% indicates oversold condition.

Figure 4. Relative Strength Index



C. Results

Comparison of execution time of algorithms on different CPU and GPU hardware platforms is done to calculate overall speedup metrics.

To calculate these parameters, the following formulae has been used.

$$\text{Speedup}_{\text{ Overall}} = \frac{1}{\left((1-x)+\left(\frac{x}{y}\right)\right)}$$

Where x= fraction $_{\text{enhanced}}$
y= speed $_{\text{enhanced}}$

Table 1. Comparison of execution times on different hardware platforms

	Average time CPU needed(sec)	Average time GPU needed(sec)	Speed up enhanced	Fraction enhanced	Overall Speedup
K-means	1.40	1.09	1.17	0.85	1.19
real time decision	4.00	2.50	1.6	0.625	1.31

2. STOCK Prediction.

Data fed to the K-means algorithm show satisfactory results about what to do with particular stock at a particular time. Choosing random value as centroid among the vector then classifying data into 3 clusters which finally results in best possible centroid values.

Figure 5. Data is classified into 3 clusters over several iterations

```
Iteration 7: centroid 0: 1368.412231  cluster_size: 64
Iteration 7: centroid 1: 1381.779297  cluster_size: 35
Iteration 7: centroid 2: 1359.605835  cluster_size: 44
Iteration 8: centroid 0: 1368.412231  cluster_size: 64
Iteration 8: centroid 1: 1381.779297  cluster_size: 35
Iteration 8: centroid 2: 1359.605835  cluster_size: 44
Iteration 9: centroid 0: 1368.412231  cluster_size: 64
Iteration 9: centroid 1: 1381.779297  cluster_size: 35
Iteration 9: centroid 2: 1359.605835  cluster_size: 44
```

According to centroid and technical indicators of each clusters after each time frame three options are provided

Buy,
Sell or
Do Nothing.

Figure 6. Decision after each time stamp

```
2019-10-15 14:55:00     Nothing
2019-10-15 15:05:00     Buy
2019-10-15 15:15:00     Buy
2019-10-15 15:25:00     Nothing
2019-10-16 09:15:00     Nothing
2019-10-16 09:25:00     Sell
2019-10-16 09:35:00     Nothing
```

VI. CONCLUSION

In this paper, an algorithm is introduced based on a parallel computing framework on CUDA platform for reduction of time and k means algorithm for clustering of stock. A dataset of particular Stock is fed into K-means algorithm. And by using that it has been attempted to achieve a real time trading simulation. This applied algorithm with the combination of CUDA is proven to work both efficiently and effectively to trade Reliance Industries Stock. Finally, the CUDA parallel computing platform reduces the loading time of OHLC values and calculation time of technical indicators as shown through the comparative chart in this paper.

Vector of simple moving average is taken into consideration, then three random centroid values are decided for 3 Clusters. As per k-Means algorithm, the simple moving average values are fed to these 3 clusters that results in adjusting centroid values according to simple moving average (SMA) vector. Doing it over several iterations results in best centroid values and each cluster contributes to the decision, whether to buy stock, sell it or stay neutral.

VII FUTURE SCOPE

This paper has only considered the single stock but in future it can analyze the model behavior with multiple stocks or even predict it for all stocks present in share market. Further it can also suggest a portfolio of stock that depends on the clustering based unsupervised learning to maximize the user profit reduce the risk in trading. To increase accuracy, the twitter sentiment analysis can be added and by merging the results of technical indicators and twitter sentiment analysis can increase the accuracy of proposed model. Form real-time applications, web app can be developed to improve the user experience. In future, the study of volatility behavior of stocks can be developed by varying the number of clusters.

REFERENCES

[1] S. Yutong and H. Zhao, "Stock selection model based on advanced AdaBoost algorithm," 2015 7th International Conference on Modelling, Identification and Control (ICMIC), Sousse, 2015, pp. 1-7, doi: 10.1109/ICMIC.2015.7409386.

[2] S. Yutong and H. Zhao, "Stock selection model based on advanced AdaBoost algorithm," 2015 7th International Conference on Modelling, Identification and Control (ICMIC), Sousse, 2015, pp. 1-7, doi: 10.1109/ICMIC.2015.7409386.

[3] He, HX & Chen, Jie & Jin, Huidong & Chen, Shu-heng & Wang, Paul & Kuo, TW. (2007). Trading Strategies Based on K-means Clustering and Regression Models. 10.1007/978-3-540-72821-4_7

[4] Wu, B., Ai, H., Huang, C., & Lao, S. (2004, May). Fast rotation invariant multi-view face detection based on real adaboost. In Automatic Face and Gesture Recognition, 2004. Proceedings. Sixth IEEE International Conference on (pp. 79-84). IEEE.

[5] Chu, F., & Zaniolo, C. (2004). Fast and light boosting for adaptive mining of data streams. In Advances in knowledge discovery and data mining (pp. 282-292). Springer Berlin Heidelberg.

[6] West, D., Dellana, S., & Qian, J. (2005). Neural network ensemble strategies for financial decision applications. Computers & operations research, 32(10), 2543-2559.

[7] Nanda, S.R. & Mahanty, Biswajit & Tiwari, Manoj. (2010). “Clustering Indian stock market data for portfolio management” Expert Syst. Appl. 37. 8793-8798. 10.1016/j.eswa.2010.06.026.

[8] Huang, K., & Thulasiram, R. K. (2005, May). Parallel algorithm for pricing American Asian options with multi-dimensional assets. In null (pp. 177-185). IEEE.

[9] Gavrishchaka, V. V. (2006). Boosting-based frameworks in financial modeling: Application to symbolic volatility forecasting. Advances in Econometrics B, 20, 123.

[10] Tamal Datta Chaudhuri, Indranil Ghosh “Using Clustering Method to Understand Indian Stock Market Volatility”, Communications on Applied Electronics (CAE) – ISSN : 2394-4714 Foundation of Computer Science FCS, New York, USA Volume 2 – No.6, August 2015.

[11] Cortés, E. A., Martínez, M. G., & Rubio, N. G. (2007). Multiclass corporate failure prediction by Adaboost. M1. International Advances in Economic Research, 13(3), 301

[12] Alfaro, E., García, N., Gámez, M., & Elizondo, D. (2008). Bankruptcy forecasting: An empirical comparison of AdaBoost and neural networks.Decision Support Systems, 45(1), 110-122.

[13] Sun, J., Jia, M. Y., & Li, H. (2011). AdaBoost ensemble for financial distress prediction: An empirical comparison with data from Chinese listed companies.Expert systems with applications, 38(8), 9305-9312.

[14] Ashish Sharma, Dinesh Bhuriya, Upendra Singh. "Survey of Stock Market Prediction Using Machine Learning Approach", ICECA 2017

[15] Arce, Paola & Maureira-Fredes, Cristián & Bonvallet, Roberto & Fernández, César. (2012). Forecasting high frequency financial time series using parallel FFN with CUDA and ZeroMQ. 2012 9th Asia-Pacific Symposium on Information and Telecommunication Technologies, APSITT 2012.

[16] Hendricks, Dieter et al. “Detecting intraday financial market states using temporal clustering.” Quantitative Finance 16 (2016): 1657 - 1678.

[17] Dongarra, J., Paprzycki, M., & Wasniewski, J. (2010). Parallel Processing and Applied Mathematics. R. Wyrzykowski (Ed.). Springer .