Comparative Analysis of Stock Market Prediction System using SVM and ANN

Himanshu H. Shrimalve
B.E. Computer Engineering
Department of Computer Engineering
NDMVPS's KBT COE, Nashik, Maharashtra

Sopan A. Talekar
Assistant Professor
Department of Computer Engineering
NDMVPS's KBT COE, Nashik, Maharashtra

ABSTRACT
Support vector machines (SVM) and artificial neural networks (ANN) are machine learning methods that find a wide range of applications both in the field of engineering and social sciences. Here, Artificial Neural Networks (ANN) and Support vector machines (SVM) are employed to predict stock market daily trends: ups and downs. The purpose is to study the variation in certain parameters like accuracy, time efficiency of both classifiers (ANN and SVM) on similar datasets in predicting stock market daily trends. In this study, different SVM and ANN models for the problem of stocks prediction which provide maximum accurate results have been applied on different combinations of data sets which obtained from the historical data of various companies and comparative analysis has been presented. The findings show that SVM and ANN models give meaningful performance results for the stock investment.

Keywords
Machine Learning, SVM, OAA-SVM, ANN, OAA-ANN, nseindia.

1. INTRODUCTION
The Indian stock market is considered to be one of the earliest in Asia, which is in operation since 1875. However, it remained largely outside the global integration process until 1991. Stock market prediction is the act of trying to determine the future value of a company stock or other financial instruments traded on an exchange. The successful prediction of a stock's future price could yield significant profit. The efficient market hypothesis suggests that stock prices reflect all currently available information and any price changes that are not based on newly revealed information thus are inherently unpredictable. We are using Technical analysis technique to enhance investors for trading and for comparative study. It is a statistical technique using an opening price, a closing price and a volume in each day for stock trading. This technique can provide a trading signal (buying or selling) to investors. Artificial Intelligence (AI) which is a technique to increase intelligence on the information system has been developed continuously. Machine Learning (ML), which is an AI tool, can be employed to learn and recognize related data patterns on the classification problems. Currently, there are several types of ML models used such as Support Vector Machines (SVM), Artificial Neural Network (ANN). The models have used the pre-processed data set of closing value of NSE and BSE India. First, we will take the dataset from NSE & BSE India official website [1] which contains historical stock information of various industries. SVM and ANN will be trained and tested using historical data values and then the real time values will be provided on which the ML models predict the trends for that company and helping Investors for making decision. We have used two ML technique for comparative study of SVM and ANN in stock market prediction research field.

2. RELATED WORK
In day to day life many peoples make invest in stocks, however some people have not got sufficient knowledge about stock due to which people have lost their money in stocks. There are many myths related about investing in stock. To give people sufficient knowledge and analysis of stock many researchers had contributed their knowledge and came up with the solution.

Radu Iacomin [9], proposed a system "Stock Market Prediction" which uses various machine learning algorithm such as Support Vector Machine with feature selection algorithm however this technique is suitable for only single class classification. Sabaithip Boonpeng, Piyasak Jeatrakul [10] proposed a system in which OAA-Neural Network is used for classifying the stock data into 3 different classes such as, buying data, selling data and holding data. However, there are several limitations that, training time required for neural network is more, it required large memory to store the data, and hence neural network is not suitable for a large dataset.

Binoy B. Nair, et al. [4] proposed a system using hybrid Decision tree and Neuro Fuzzy which has required large searching time and memory.

QIU Mingyue., et al. [8] proposed a system using hybrid Genetic Algorithm and Artificial Neural Network which has high convergence rate.

To overcame the drawbacks of all above system Xiaowei Yang., et al. proposed OAA-SVM technique for multiclass classification using SVM [6].

Xiaowei Yang, et al. proposed a short term prediction system based on echo state networks (ESN), which outperforms other conventional neural networks in some cases. It included principle component analysis (PCA) to filter noise in data department and choose appropriate parameters [3].

Chen, Leung, and Daouk (2003) used probabilistic neural network (PNN) to predict the direction of Taiwan stock index return. They reported that PNN has higher performance in stock index than generalized methods of moments-Kalman filter and random walk forecasting models [2]. Amin Hedayati., et al. studied the ability of neural network (ANN) in forecasting the daily NASDAQ stock exchange rate. The methodology used in this study considered the short-term Historical stock prices as well as the day of week as inputs. The model outputs show that there is no distinct difference between the prediction ability of the four and nine prior working days as input parameters [7]. It is summarized in the Table 1.

Table 1. Related Work

Title	Year	Technology Used	Limitations
Stock Market Prediction[9]	IEEE (2016)	Support Vector Machine	Single class classification technique.
Decision Support System for Investing in Stock Market by using OAA-Neural Network[10]	2016	OAA-Neural Network	More training time required. Large memory to store data. Not suitable for large datasets.
A Stock market trend prediction system using a hybrid decision tree-neuro-fuzzy system[4]	IEEE (2016)	Hybrid Decision tree and Neuro-Fuzzy	Large searching time and large Memory required.
Application of the Artificial Neural Network in predicting the direction of stock market index[8]	IEEE (2016)	Hybrid Genetic Algorithm and Artificial Neural network	High convergence rate
Short-term stock price prediction based on echo state networks[3]	2009	Novel Neural Network –echo state network (ESN) with principle component analysis (PCA).	Suitable for stocks that have lasting obvious trend and fluctuate a little.

3. THEORETICAL FRAMEWORK:

3.1 Support Vector Machine (SVM):

Support Vector Machine (SVM) is Supervised Learning technique of Machine Learning (ML). In SVM, support vectors are shown in figure 2. It analyzes the data and recognize the patterns, which are used for regression and classification analysis. SVM training algorithm builds a model that assigns new examples to one category or the other, making it a non-probabilistic binary linear classifier. [11] An SVM model is the representation of the examples as points in space, mapped so that the examples of the separate categories are divided by a clear gap that is as wide as possible. New examples are then mapped into that same space and predicted to belong to a category based on which side of the gap they fall.

SVM builds hyper-plane or set of hyperplanes, which can be used for regression and classification as shown in figure 1. Good separation is done by the hyper plane that also has the largest distance, if margin is higher generalization of error of classifier is lower. [11]

3.1.1 Mathematical Equations

For 2D dataset let,

D is the dataset containing $\{x_i, y_i\}$ number of tuples where,

x_i = Set of attributes/tuples

y_i = Set of classes

As, dataset is 2D, y_i = +1/–1

- Equation of hyperplane:

$$\boldsymbol{w}.\boldsymbol{x} + \boldsymbol{b} = \mathbf{0} \qquad \text{(Eq.1)}$$

Where,

w = Weight vector (Distance of attribute from the hyperplane)

x = No of Attribute

b = Scalar/ Bias [11]

Equation of attributes lies on hyperplane:

$$W0 + W1.X1 + W2.X2 = 0 \qquad \text{(Eq.2)}$$

Equation of attributes lies above the hyperplane

$$H1: W0 + W1.X1 + W2.X2 \geq 0. \qquad \text{(Eq.3)}$$

Equation of attributes lies below the hyperplane: [11]

$$H2: W0 + W1.X1 + W2.X2 \leq -1 \qquad \text{(Eq.4)}$$

Combining equation of attribute above hyperplane (H1) and below the hyperplane (H2), we have obtained following equation [11]

$$Y_i(W0 + W1.X1 + W2.X2) \forall I \qquad \text{(Eq.5)}$$

Maximum Marginal Hyperplane (MMH) given by: [11]

$$\frac{2}{||W||} \text{ , where } ||W| Ecledian\ norm\ of\ W \qquad (Eq.6)$$

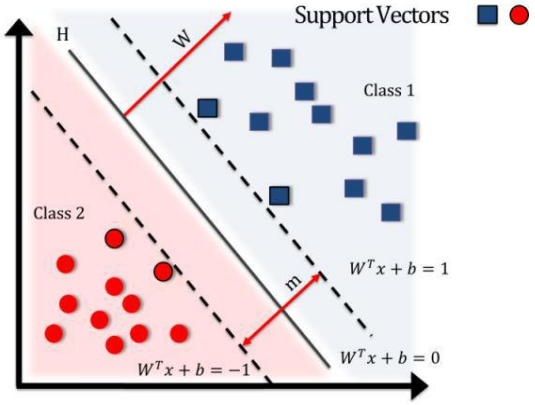

Figure 1. Support Vector Machine (SVM) [5]

3.2 Artificial Neural Network:

The term 'neural network' has its origins in attempts to find mathematical representations of information processing in biological systems (Bishop, 1995).

An artificial neuron (AN) is a model of a biological neuron (BN). Each AN receives signals from the environment, or other ANs, gathers these signals, and when fired, transmits a signal to all connected ANs. Figure 2 is a representation of an artificial neural network. Input signals are inhibited or excited through negative and positive numerical weights associated with each connection to the AN. The fire of an AN and the strength of the exiting signal are controlled via a function, referred to as the activation function. The AN collect all

incoming signals and computes a net input signal as a function of the respective weights. The net input signal serves as input to the activation function which calculates the output signal of the AN (EngelBrecht, 2007).

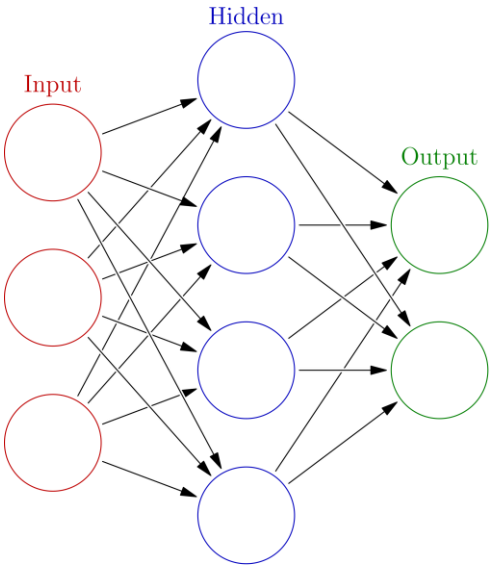

Figure 2. Artificial Neural Network (ANN)

3.2.1 Components of Artificial Neural network:

A neuron with label *j* receiving an input $p_i(t)$ from predecessor neurons consists of the following components

i. an activation $a_j(t)$, *depending on discrete time parameter*,
ii. possibly a threshold Θ_j, which stays fixed unless changed by a learning function
iii. an activation function f that computes the new activation at a given time $t+1$ *from* $a_j(t)$, Θ_j and the net input $p_j(t)$ giving rise to the relation

$$aj(t+1) = f(aj(t), pj(t), \Theta j) \quad \text{(Eq.7)}$$

iv. and an output function f_{out} computing the output from the activation

$$\text{oj(t)} = fout(aj(t)) \quad \text{(Eq.8)}$$

Often the output function is simply the Identity function.

v. An *input neuron* has no predecessor but serves as input interface for the whole network. Similarly an *output neuron* has no successor and thus serves as output interface of the whole network.
vi. Connections and weights:

The network consists of connections, each connection transferring the output of a neuron *i* to the input of neuron *j*. In this sense *i* is the predecessor of *j* and *j* is the successor of *i*. Each connection is assigned a weight w_{ij}.

vii. Propagation function:

The propagation function computes the input $p_j(t)$ to the neuron *j* from the outputs $o_i(t)$ of predecessor neurons and typically has the form

$$p_j(t) = \sum_i o_i(t) w_{ij} \quad \text{(Eq.8)}$$

3.3 Technical Indicators Used:

3.3.1 Relative Strength Index (RSI)

RSI indicates trading signal based on the relationship of stock price in the current day and stock price in the past period. RSI value can be calculated by the following formula: [11]

$$RSI = 100 - [100/1 + RSI] \quad \text{(Eq.10)}$$

$$RSI = \frac{Average\ of\ 14\ days\ closes\ up}{Average\ of\ 14\ days\ closes\ up} \quad \text{(Eq.11)}$$

3.3.2 Stochastic Oscillatory Technique (SOT)

Stochastic Oscillator technique compares the current close price of a stock with its price range over a period in the past. It uses two lines for generating trading signal, which are %K line and %D line. These lines are calculated by the following formula: [11]

$$K\%line = 100 - \frac{(Recent\ closes - Lowest\ low(n))}{(Highest\ High(n) - Lowest\ Low)} \quad \text{(Eq.12)}$$

$$D\%\ line = 3\ period\ moving\ average\ of\ \%K\ line \quad \text{(Eq.13)}$$

Stochastic technique indicates trading signals when %K line and %D line is across each other above 80 or below 20. The selling signal is generated when %K line and %D line across each other above 80 while the buying signal is indicated when %K line and %D line across each other below 20. [11]

3.3.3 Moving Average Convergence and Divergence (MACD)

MACD technique is proposed by Gerald Appel. This technique can indicate trading signals (buying and selling) and also the trends of a stock price (upward trend, downward trend and sideway trend). MACD technique indicates trading signal based on the difference of two moving average lines. These are 12 days and 26 days. MACD value is calculated by the following formula: [11]

$$MACD\ Line = EMA(12) - EMA(26) \quad \text{(Eq.14)}$$

MACD technique indicates trading. When MACD crosses over zero line, the buying signal is indicated and the trend of stock is starting an upward trend. While MACD crosses under zero line, the selling signal is generated and the downward trend is beginning.

3.3.4 Rate of Change (ROC)

The Rate of Change Indicator (ROC) is a momentum oscillator. It calculates the percentage change in price between periods. ROC takes the current price and compares it to a price "n" periods ago.

$$ROC = \left[\frac{(CurrentClose - Close\ n\ periods\ ago}{(Close\ n\ periods\ ago)}\right] \times 100 \quad \text{(Eq.15)}$$

Where,

n= A user defined number.

3.3.5 Exponential Moving Average (EMA)

Moving Average is a price based, lagging (or reactive) indicator that displays the average price of a security over a set period of time. The major difference with the EMA is that old data points never leave the average. There are 3 steps to calculate the EMA.

i. Calculate the SMA

$$SMA = Period\ Values/Number\ of\ Periods \quad \text{(Eq.16)}$$

ii. Calculate the Multiplier

$$Multiplier = (2/(Number\ of\ Periods + 1)) \quad \text{(Eq.17)}$$

iii. Calculate the EMA

$$EMA = \{Close - EMA(previousdaay)\} \times multiplier + EMA(previousday) \quad \text{(Eq.18)}$$

4. RESULTS AND DISCUSSION:

In this study the data from 2016-2018 periods in EQUITY index were used. 6 independent variables from the historical data of stock prices have been identified and calculated, as the dependent variable, class variable have been labelled according to growth and fall in stock values as "1" and "0". Class label "1" shows the growth in price and "0" indicates fall in price according to previous value.

Based on the mention data, the results of SVM and ANN models have been compared using various kernels and partition schemes.

4.1 Support Vector Machine (SVM):

SVM model has been developed in Python 3.6 using Sci-kit learn library which is a powerful machine learning tool. Dependent and independent dataset are divided into testing and training datasets using cross-validation splitting method. Datasets are divided into 70% for training dataset and remaining 30% for testing dataset. Changing the split ration also changes the outcome results, that's why it is necessary to divide datasets in proper ratio to get maximum accuracy. In the Table 2 below, I have given the changes in results according to split ratio.

Table 2. Ratio-Accuracy Relation

Sr. No.	Model	Dataset Divide Ratio	Accuracy (%)
1	SVM	7:3	71.1267%
2	SVM	13:7	64.8484%
3	SVM	3:1	64.4067%
4	SVM	4:1	66.3157%

Also, accuracy of SVM classification model depends on the kernel selected. There are multiple kernels are supported by SVM which can be used as per requirements. In this case study we have used "linear" model for classification but we have studied all models and compared their results for gaining maximum accuracy which was achieved by "linear" kernel. Table 3 shows the accuracy level achieved by each kernel. Default kernel type while creating a classifier instance is "rbf (Radial Basis Function)".

Accuracy of model can be computed by matching results in testing dataset with predicted results.

Table 3. SVM Kernel-Accuracy relation

Sr. No.	SVM Kernel	Accuracy(%)
1	Rbf	64.67%
2	Linear	71.83%
3	Sigmoid	61.33%
4	Poly	65.33%

A very efficient tool is provided by sklearn library for this purpose. A confusion matrix can be created to know the matched and unmatched results which can lead to compute accuracy.

Confusion matrix consist of 4 elements, which are

1. False-False: Actual value is False and predicted as False.
2. False-True: Actual value is False and predicted as True.
3. True-False: Actual value is True and predicted as False.
4. True-True: Actual value is True and predicted as True.

So, we have computed the results on "linear" kernel with 7:3 dataset dividing ratio with 2 years data of "SBIN" company. Following Figure 3 shows confusion matrix results for same.

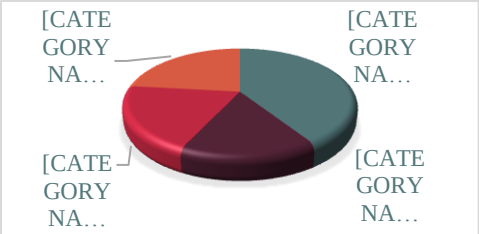

Figure 3. SVM Confusion Matrix Results

4.2 Artificial Neural Network:

For developing Neural Network 3 libraries are required which are Theano, Keras and Tensorflow in python 3.6.

Keras needs to run on tensorflow environment, which is the essential tool for creation of layers i.e. input layer, hidden layer and output layer.

Activation functions are really important for an Artificial Neural Network to learn and make sense of something really complicated and Non-linear complex functional mappings between the inputs and response variable. They introduce non-linear properties to our Network. Their main purpose is to convert an input signal of a node in a A-NN to an output signal. That output signal now is used as an input in the next layer in the stack.

Types of Activation Functions:

i. Sigmoid Function:

It is an activation function of form

$$f(x) = \frac{1}{1+e^{-x}} \quad \text{(Eq.19)}$$

Its Range is between 0 and 1. It is a S—shaped curve. It is easy to understand and apply. Its output isn't zero centered. It makes the gradient updates go too far in different directions. 0 < output < 1, and it makes optimization harder.

ii. Hyperbolic Tangent Function- Tanh**:**

Its mathematical formula is

$$f(x) = \frac{1-e^{-2x}}{1+e^{-2x}} \quad \text{(Eq.20)}$$

Now its output is zero centered because its range in between -1 to 1 i.e. -1 < output < 1. Hence optimization is easier in this method hence in practice it is always preferred over Sigmoid function. But still it suffers from Vanishing gradient problem.

iii. ReLu- Rectified Linear Function:

It has become very popular in the past couple of years.

It was recently proved that it had 6 times improvement in convergence from Tanh function. It's just

$$R(x) = max(0, x)\ i.e.\ if\ x < 0, R(x) = 0 \quad \text{(Eq.21)}$$

$$and\ if\ x >= 0, R(x) = x \quad \text{(Eq.22)}$$

Hence as seeing the mathematical form of this function we can see that it is very simple and efficient. A lot of times in Machine learning and computer science we notice that most simple and consistent techniques and methods are only preferred and are best.

In this study we are using rectifier function for hidden layers and sigmoid function for output layer it gives the probabilistic output which we need to round off to nearest value to get actual '0' or '1' value, where '0' is used for fall in price and '1' is used for growth in price.

Firstly, random weights are assigned to neurons from input layer to hidden layer from left to right which is known as forward propagation. The neurons are activated in a way that the impact of each neuron's activation is limited by the weights. Then comparing the predicted results with actual result, generated error is calculated. From that error is back-propagated and weights are updated to the neurons according to how much they are responsible for error.

In ANN also the dataset is divided into 2 parts i.e. training dataset and testing dataset. Further training dataset is divided into number of batches of equal data samples.

Table 4. Effect of Batch size and Epochs on Accuracy

Sr. No.	Batch Size	No. of Epochs	Max. Training Accuracy (%)	Testing Accuracy (%)
1	30	50	65.42%	71.14%
2	15	70	66.57%	70.46%
3	50	100	65.99%	70.46%
4	75	30	51.01%	55.70%

These batches are trained to model multiple times i.e. in multiple epochs. This is generally referred as "Batch Learning". Batch size and no of epochs are also responsible for model accuracy.

As shown in Table 4, batch-size and number of epochs made a major impact on model accuracy. Batch-size and number of epochs is needed to decide very specifically by studying about the results. Initially we selected the batch-size of 30 and 50 number of epochs. As we decreased the batch size and increased the no of epochs i.e. number of iterations, it didn't make much difference but it required more time. Then when batch size is increased and number of epochs are also increased training accuracy is not differed but testing accuracy is decreased a bit and time required for execution is increased which is not efficient. After this when batch size is increased even more and number of epochs are decreased to 30 epochs only it made a high difference in training and testing accuracy. Training and testing accuracies decreased in range of 50 to 55 percentile which is very low. From above observation, conclusion about Batch training can be derived as epochs are very responsible for error reduction, hence number of epochs must be high but in limited range after which accuracy didn't changes very much.

The overall accuracy of ANN model can be derived from confusion matrix. Following results are achieved on "SBIN" stock from 2016-2018 with rectifier activation function for hidden layers and sigmoid function for output layer.

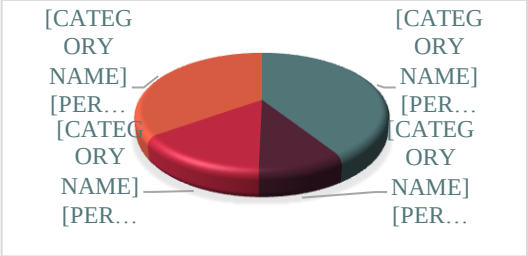

Figure 4. ANN Confusion Matrix Results

Below are the sample test results are given for "SBIN" stock, historical data used for prediction is from 2016-05-30 to 2018-05-25. Technical Indicators used for predictions were RSI (Relative Strength Index), EMA (Exponential Moving Average), MACD (Moving Average Convergence Divergence), ROC (Rate of Change). Dataset was divided into 70% for training and 30% for testing. For NN Batch-size was 30 and 50 epochs were used. Table 5 shows the sample test results for SVM model using linear kernel and Table 6 shows the sample test results for ANN mode.

Table 5. Sample test results of SVM model

Date	Open Price	Close Price	Actual Result	Predicted Result
2018-05-11	247.6	250.35	Up	Up
2018-05-14	249.95	253.6	Up	Down
2018-05-15	253	248	Down	Down
2018-05-16	245.5	243.1	Down	Down
2018-05-17	244.2	242.7	Down	Up
2018-05-18	243	238.85	Down	Up
2018-05-21	243.55	244.45	Up	Up
2018-05-22	243.65	253.9	Up	Up

Table 6. Sample test results of ANN model

Date	Open Price	Close Price	Actual Result	Predicted Result
2018-05-11	247.6	250.35	Up	Up
2018-05-14	249.95	253.6	Up	Down
2018-05-15	253	248	Down	Down
2018-05-16	245.5	243.1	Down	Down
2018-05-17	244.2	242.7	Down	Up
2018-05-18	243	238.85	Down	Down
2018-05-21	243.55	244.45	Up	Up
2018-05-22	243.65	253.9	Up	Up

5. CONCLUSION

The results of both the models are very close to each other. It concludes that SVM and ANN both are efficient machine learning models for stock market prediction. As ANN requires more time in case of more number of epochs as compared to SVM but it is negligible as ANN give somewhat more accurate results.

However, SVM and ANN models gives reasonable results for stock prediction problem by adding technical indicators, but more efficient and robust models can be designed for further research.

In future, recent machine learning strategy i.e. deep learning will be used to improve the performance of the system and results will be compared with the neural network approach. In this paper comparative analysis is done only for prediction of UP or DOWN trend in stock market. In the future scope comparative analysis of stock price prediction for upcoming time period will be done using deep learning and other ML techniques.

6. REFERENCES

[1] www.nseindia.com (last visited on 22 Feb 2018)

[2] Chen, A. S., Leung, M. T., & Daouk, H. (2003). "Application of neural networks to an emerging financial market: Forecasting and trading the Taiwan Stock Index." Computers & Operations Research, 30, 901–923.

[3] Xiaowei Lin , Zehong Yang, Yixu Song (2009) "Short-term stock price prediction based on echo state networks" Expert Systems with Applications 36, 7313–7317.

[4] Binoy B. Nair, N. Mohana Dharini, V.P. Mohandas, "A Stock market trend prediction system using a hybrid decision tree-neuro-fuzzy system", IEEE xplore, 16-17 oct 2010.

[5] TİMOR, M & Dincer, Hasan & Emir, Şenol. (2012), "Performance comparison of artificial neural network (ANN) and support vector machines (SVM) models for the stock selection problem: An application on the Istanbul Stock Exchange (ISE)-30 index in Turkey." African Journal of Business Management. 6. 1191-1198.

[6] Xiaowei Yang, Qiaozhen Yu, Lifang He, Teng Jiao Guo, "The one-against-all partition based binary tree support vector machine algorithms for multi-class classification", School of Computer Science and Engineering, South China University of Technology, Guangzhou 510641, PR China, Neurocomputing volume 113, August 2013.

[7] Amin Hedayati Moghaddama, Moein Hedayati Moghaddamb, Morteza Esfandyaric, "Stock market index prediction using artificial neural network", Journal of Economics, Finance and Administrative Science 2016, 89–93.

[8] QIU Mingyue, LI Cheng, Song Yu, "Application of the Artificial Neural Network in predicting the direction of stock market index", Department of System Management, Fukuoka, Japan, 2016.

[9] Radu Iacomin, "Stock Market Prediction", Faculty of Automatic Control and Computers University POLITEHNICA of Bucharest, Romania, 2016, pp 200-205.

[10] Sabaithip Boonpeng and Piyasak Jeatrakul, "Decision Support System for Investing in Stock Market by using OAA-Neural Network", School of Information Technology Mae Fah Luang University Muang, Chiang Rai, IEEE digital xplore , 11 April 2016.

[11] Himanshu Shrimalve, Shreya Sulakhe, Madhavi Sontakke, Mayuri Thakare and Sopan Talekar, "Decision Support System for Investment in Stock Market using OAA-SVM" MVP Journal of Engineering Sciences, Vol 1(1), DOI:10.18311/mvpjes/2018/v1i1/18256, June 2018.