Journal of Engineering and Applied Sciences 13 (Special Issue 5): 4794-4801, 2018
ISSN: 1816-949X

A New Deep Neural Network Regression Predictor Based Stock Market

Eman Al-Shamery and Ameer Al-Haq Al-Shamery
Department of Software, Faculty of Information Technology, University of Babylon, Hillah, Iraq

Abstract: Stock Market (SM) prediction is an interesting financial topic that has attracted the attention of researchers for the last years. SM dataset is a chaotic, non-linear, dynamic, non-stationary, noisy in behavior and quite difficult. The aim of this study is to predict the future SM price. The proposed system consists of three major stages, the first stage is preprocessing data that focuses on preparing the data for mining process. It includes features extraction; transform nominal to numeric and interpolation. The second stage involves building new prediction model called: Deep Neural Network (DNN) Regression Predictor (RP) that it is used for price prediction. The DNN-RP consists of three layers (technical indicators generation layer, standardization regression layer and regression predictor layer). In third stage, the evaluation has been performed depending on popular measures of prediction and 10-Cross Validation (CV). The DNN-RP is evaluated by Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) measures. The system is applied on the three global datasets of SMs (Dow, S and P500 and NASDAQ) in addition to one local SM specifically is (Bank of Baghdad). Finally, the proposed system has been compared with other popular methods. The DNN-RP is compared with K-Nearest Neighbor (KNN) and Support Vector Machine (SVM)), the results of the proposed system are better for all evaluation parameters. The best MAE and RMSE values of DNN-RP have reached to 0.0051 and 0.007, respectively.

Key words: Stock market prediction, technical indicators, Deep Neural Network (DNN), Regression Predictor (RP), K-Nearest Neighbor (KNN), layers

INTRODUCTION

SM analysis have occupied broad area for researchers and investors. The price prediction of SM is one of an interesting field at present because it is an effective part of any country's economy. It plays a significant role in the industry which in the end effect on country's economy growth (Naeini *et al*., 2010). The SM is a public place where companies allow to trading of moneys through the sale or purchase of shares and stocks after determining the price agreed upon in advance (Chavan and Patil, 2013).

Data Mining (DM) achieves suitable knowledge from data using machine learning techniques and statistical methods (Pang-Ning *et al*., 2005). DM contents three main phases: the data is prepared to mining process then applying machine-Learning techniques for knowledge extraction and finally the results will process in clear form to be helpful for decision-making. Mostly, DM is used for two purposes; distinguishing hidden patterns in data or predicting unknown values of interested features from known values of other features (Han and Kamber, 2006).

The prediction of SM price is the process to determine the future value. Predicting the stocks prices can be done in two way, first: fundamental analysis at most be determined by mathematical data of a company (Robert *et al*., 2001). Second: technical analysis at most be determined by Technical Indicators (TIs) and machine learning, the analyst using numerical factors include daily ups and downs, volume of stock, tendency pointers, highest and lowest prices of a day, directories, simple moving average and others. Analysts have attempted to discover some accurate arithmetic model which can allocate inputs to predict the actual output (Yefimochkin, 2011). The datasets of SM such as other financial datasets have many properties which make prediction process more difficult (Dean, 2014).

Literature review: By Shen *et al*. (2012), the researcher implement new prediction algorithm that exploits the temporal correlation among global SMs and various financial products to predict the stock trend with the utility of SVM. Numerical results indicated prediction accuracy of 74.4% for NASDAQ, 76% for S and P500 and 77.6% for DOW.

By Nemes and Butoi (2013) present series of Neural Networks (NNs) which are designed for stock exchange rates forecasting. In order to predict short-time price fluctuations a multistep ahead strategy was used. Later,

Corresponding Author: Ameer Al-Haq Al-Shamery, Department of Software, Faculty of Information Technology, University of Babylon, Hillah, Iraq

the findings of their study can be integrated with an intelligent multi-agent system model for helping users I n the decision making process of buying or selling stocks.

By Yetis *et al*. (2014), the researchers have used NNs with the set of input parameters of the market to predict NASDAQ's stock price. The researchers use feed forward NNs. Regression technique is employed to check the network performance. The generated strategies for validation the outputs, training and test cases. The NN was trained using input data of SM price in period between 2012 and 2013.

Panigrahi and Mantri (2015) introduce a comparative study between discrete features and numerical features as input to C4.5 for forecasting. Where the historical data and twelve technical indicators are calculated over numerical dataset of two different SMs as input to classification model. The researchers have founded that a text based decision tree model leads to better accuracy than the usual decision tree.

By Chen and He (2015), a comparative study between Back Propagation BPNN and SVM. for prediction is introduced. Additionally, the researchers applied Piecewise Linear Representation (PLR). The PLR is used for checking of watershed in their studies because it is good for mining worthy information from a time series. The investigation showed that prediction in SVM is better than BBNN. The to an average accuracy of SVM reached to 72.4%.

By Boonpeng and Jeatrakul (2016), the researchers classified three actions of stock movement and presented a comparative study between the One-Against-All (OAA), One-Against-One (OAO) NN and traditional NN. The fundamental analysis and the technical analysis are used for decision-making. Historical data is collected from the stock exchange of Thailand for 7 years of the period from (03/01/2007) to (29/08/2014). They found that OAANN achieve better accuracy than OAONN and traditional NN models.

By Mingyue *et al*. (2016), the researchers optimized the ANN model using Genetic Algorithms (GA) and two types of input variables using TIs are applied to predict the direction of prices of the Japanese SM index. After that, the researchers compared between type 1 and type 2 as input to model. The experimental results showed that the type 2 provide better prediction accuracy of the direction than type 1.

By Luo *et al*. (2016), this research introduced a comparative study between multiple stock price forecasting algorithms to forecast SM. They compare Principal Components Analysis (PCA) method and Genetic Algorithm (GA) to optimize the BP neural network. The experiments implemented on the Shanghai index data to make simulation. The results showed that the optimization method enhance performance of forecasting accuracy.

MATERIALS AND METHODS

Technical indicators generation: TIs could be defined as the mathematical calculations results that long or short use of term investors in their charts (on other time frames sets) (Atsalakis and Valavanis, 2009). TIs are classified as lagging and leading TIs. The lagging TIs are employed to follow the action of theprice and have less predictive abilities (Fernandez-Blanco *et al*., 2008). On the other hand, leading TIs are created to proceed the price movements of a market giving predictive abilities, i.e., the signal is generated before the reversal or new trend takes place (Fernandez-Blanco *et al*., 2008).

In this study, technical indicators are used as inputs to the model of prediction to improve the accuracy. Table 1 displays 20 TIs.

Stock market prediction techniques: The evaluation in this study is implemented by two directions. The first direction a comparative study is introduced with two popular prediction methods (K-NN and SVM).Whereas an evaluation error metrics: Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) are applied in the second direction.

K-Nearest-Neighbor: KNN is one of lazy learning type, which predicts classes of entity based on the K nearest training instances in the feature space. The K-NN classifiers works by calculating the similarity between the training objects and a specific test object. How to determine the correct value of K in this method is considered a problem.

Support Vector Machine (SVM): SVM is a supervised learning algorithm based on statistical learning theory introduced by Vapnik. It has great performance, since, it can handle a non-linear classification efficiently by mapping samples from low-dimensional input space into high-dimensional feature space with a non-linear kernel function. Unfortunately, the practical use of SVM is limited because the qualities of SVM Models heavily depend on a proper setting of SVM kernel parameters.

Proposed system: Figure 1 illustration the architecture of the proposed system for SM prediction for understanding is easy and quickly.

Table 1: Description of technical indicators

Technical indicators	Names	Description
SMA	Simple Moving Average	Simple moving average of the close price for last 30 days
WMA	Weighted Moving Average	Weighted moving average of the close price for past 30 days
EMA	Exponential Moving Average	Exponential moving average of the close price for last 30 days
AROON	Aroon	Determines whether a stock is trending or not and how strong the trend is. It is calculate depending on high and low price for last 14 days
ROC	Rate of Change	Compute rate of change relative to previous trading intervals, we use 10 days intervals
RSI	Relative Strength Index	Suggests the overbought and oversold market signal. Relative Strength index is related to 14 days max and min. avg
CCI	Commodity Channel Index	Identifies cyclical turns in stock price. It is calculate depending on high, low and close price for last 14 days
MACD	Moving Average Convergence	Moving average convergence divergence is the difference between
MINMAX	Minimum and Maximum price	Lowest and highest close values for the last 30 days
VAR	Variance	Variance of the close price for last 5 days with deviation of 1
BBANDS	Bollinger Bands	It is plotted two standard deviations away from a simple moving average
MOM	Momentum	The difference between previous close and current close for last 10
STOCH	Stochastic	The stochastic oscillator is a momentum indicator comparing the closing price of a stock market to the range of its prices over 5 days
R (%)	Williams R (%)	Determines where today's closing price fell within the range on last 10 days transaction
TRUE_RANGE	Average True Range	Shows volatility of market depending on high, low and close price
AVG	Average price	A representative measure of a range of prices that is depending on open, low, high and close price
MEDIAN	Median price	Determines median price of market depending on high and low price
WCP	Weighted Close Price	The weighted close formula calculates the average value of daily prices but gives more weight to the close price
STDDEV	Standard Deviation	A measure of the dispersion of the close price for last 5 days from its mean
TSF	Time Series Forecasting	Calculates the linear regression of 14 days price

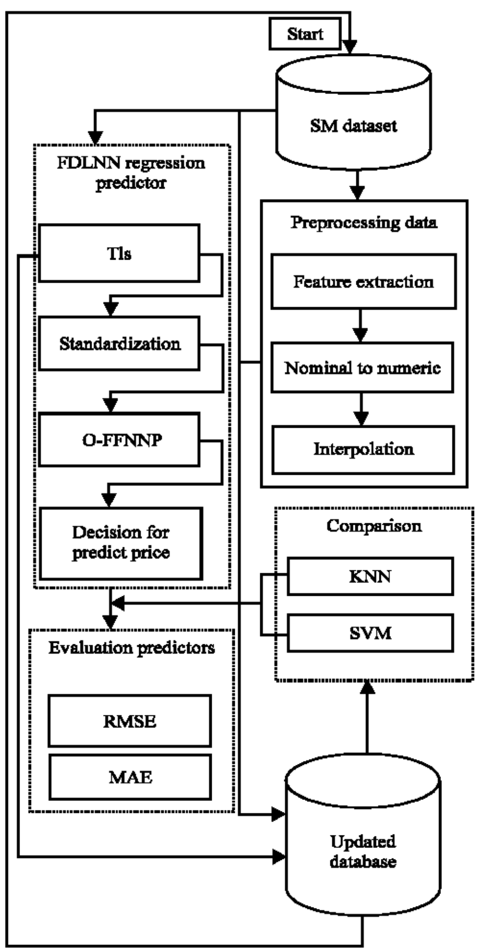

Fig. 1: The proposed system

Preprocessing stage: This stage consists of three steps include features extraction, transforming nominal to numeric and interpolation. Algorithm 1 reviews the preprocessing stage.

Algorithm 1; Preprocessing SM dataset:

Input: RDS: Raw SM Dataset
Output: DS: Processed SM Dataset
Step 1: select only standard features F from RDS
Step 2: transforming Date nominal feature (Dno) in RDS, to numeric feature by:
  Set count = 0
  For each nominal value in Dno:
    Create new empty feature (Dnu)
    Set the value of new feature (Dnu) by:
    Dnu [count] = value (concatenation (Dno [count]))
    Count = count+1
Step 3: linear regression is used to find missing price values by replacing them with existing values of prices:
    For each Feature f in F
   For each data row r in DS

$$Slop_r = \frac{\Sigma_{t=r-n}^{i}(Dnu_t - \mu Dnu_n).(f_t - \mu_{fn})}{\Sigma_{t=r-n}^{i}(Dnu_t - \mu Dnu_n)}$$

$$f_r = \mu_{fn} - slop_r * \mu_{Dnu_n}$$

Step 4: Set DS = RDS after removing Dno feature
End

Features extraction: Standard daily features (date, open, high, low, close, trading volumes and adj. close) are taken as inputs to proposed system.

Nominal to numeric: The main effectiveness of pre-pressing is preparing data for mining in proper form. The proposed system uses linear regression in the next step to address missing values and this requires

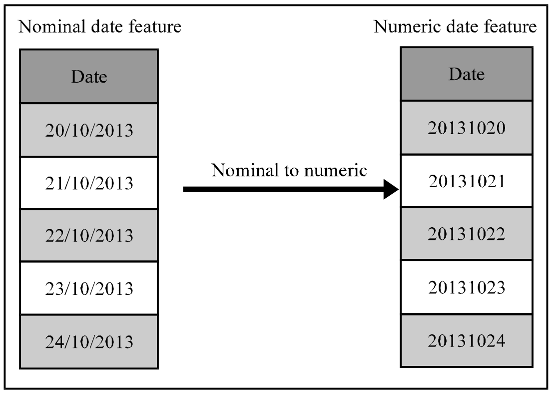

Fig. 2: Nominal to numeric

transforming nominal features of SM dataset to numerical features. Figure 2 describe the idea of transformation for example of SM data.

Interpolation: Linear Regression (LR) is employed for finding in missing values using step 3 in Algorithm 2.

Prediction stage: This stage involves building new prediction model which can successfully trade SM for a profit. This has achieved by building a certified and a perfect regression model for the SM using deep learning concept of NN where this model depends on solving the problem completely from the input until predicting the price. This model consists of three main layers (TIs generation layer, standardization regression layer and regression predictor layer) sequentially with specific objective to each layer. The topology of NN determines according to the problem given as a first step before design any NN. The topology of the suggested DNN is shown in Fig. 3.

Layer 1; Tis generation layer: The aim of this layer is to enhance the accuracy of prediction by taking the standard features (seven features) of the SM as an inputs for the network. The outputs from this layer are the twenty TIs generator from the multiplication of the Eq. 1 which represent the nets in Algorithm 3 by the weights which is shown in Table 2. These weights determine the connection between input neurons and output.

The major characteristic of this layer is each output neuron (indicator) connects only with input neurons that contribute to find this indicator. For additional details about layer 1, look at the Algorithm 3.

Table 2: The weights of layer 1

Weight	Values
W1	1/n
W2	1.00
W3	1.00
W4	100/n
W5	100
W6	100
W7	1/0.015
W8	1/n
W9	1.00
W10	1/n
W11	1.00
W12	1.00
W13	100
W14	100
W15	1.00
W16	0.25
W17	0.50
W18	0.25
W19	1.00
W20	1.00

n is the number of period time (days ago) used in each indicator

Mathematical formulas of TIs equations have been converted into (nets and weights) which a suitable formulas for Nns by applying a principle of (statistical with NNs). According to Fig. 3, the features (volume and adj. close) did not contribute to find any indicators in layer 1 but are used as input only in layer 2.

Algorithm 2; TIs generation layer:

Input: Array of standard features (X_{hk}) where h: is the number of standard features and k: is the sample's index
Output: Array of output neurons (O_{1jk}) where j: is the number of TIs
Begin
Step 1: The input neurons are formed in two dimensions according to the number of standard features represent columns and input data represent rows Additionally, the initial Weights (W) are set to constant value shows in Table 2
Step 2: For i = 1 to k:
Step 2.1: Find SMA
$I_{1i} = \Sigma_{t=i-n}^{i} X_{1t}$
$O_{11i} = W_1 * I_{1i}$
Step 2.2: Find EMA
$K1 = 2/(1+n)$
$I_{2i} = [K1 * X_{1i} - O_{12i-1}] + O_{12i-1}$
$O_{12i} = W_2 * I_{2i}$
Step 2.3: Find WMA
$I_{3i} = \Sigma_{t=i-n}^{i} X_{1t} / \Sigma_{t=1}^{n} t$
$O_{13i} = W_3 * I_{3i}$
Step 2.4: Find AROONUP
$I_{4i} = n\text{-}\max(X_{3n})$
$O_{14i} = W_4 * I_{4i}$
Step 2.5: Find ROC
$I_{5i} = X_{1i}/X_{1i-n} - 1$
$O_{15i} = W_5 * I_{5i}$
Step 2.6: Find RSI
If (Flag = false)
// Find first gain average
For t = i-n to i
If ($X_{1t} > x_{1t-1}$)
Avg Gain = AvgGain+X_{1t-1}-X_{1t}
else
AvgLoss = AvgLoss+X_{1t}-X_{1t-1}
End If

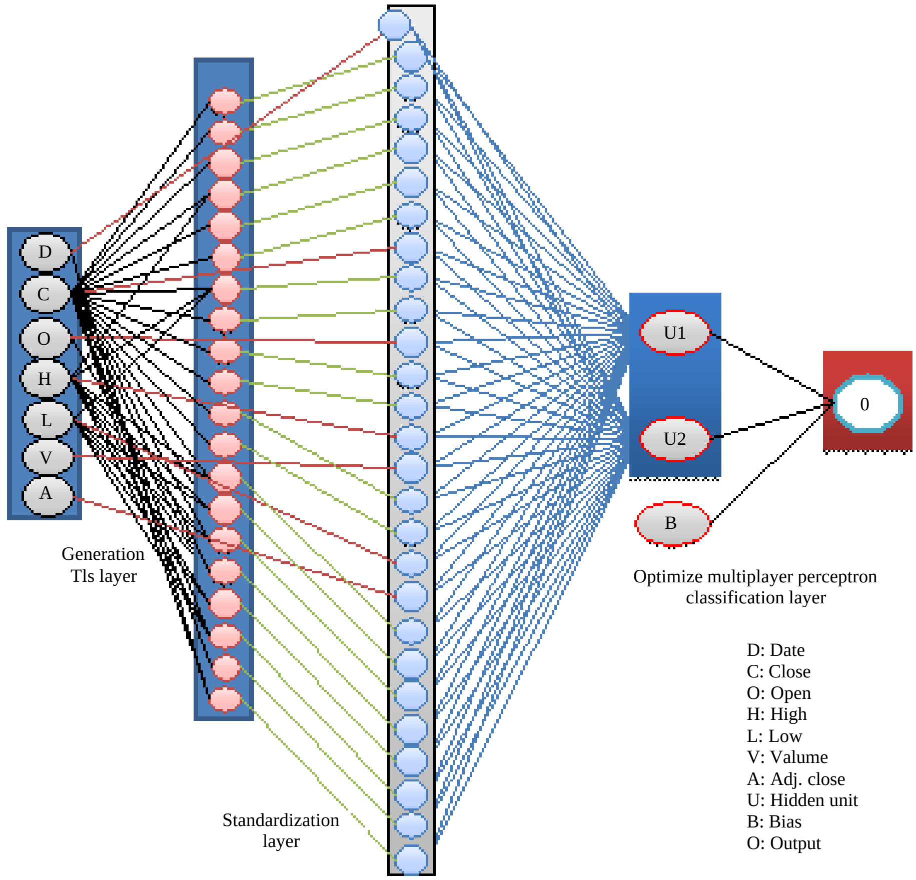

Fig. 3: The topology DNN-RP Model

End loop

Flag = true

Else

AvgGain = [PreviousAvgGain*(n-1)+CurrentGain]/n

AvgLoss = [PreviousAvgLoss*(n-1)+CurrentLoss]/n

End if

RS = AvgGain/AvgLoss

$I_{6i} = 1-[1/(1+RS)]$

$O_{16i} = W_6 * I_{6i}$

Step 2.7: Find CCI

$Tp_t = [X_{1t}+X_{3t}+X_{4t}]/3$

$\mu = \Sigma^{i}_{t=i-n} Tp_t$

$$MD = \frac{\Sigma^{i}_{t=i-n} |Tp_t - \mu|}{n}$$

$I_{7i} = [Tp_i-\mu]/ MD$

$O_{17i} = W_7 * I_{7i}$

Step 2.8: Find MACD

$K1 = 2/(1+12), K2 = 2/(1+26)$

$A = [K1 * X_{1i}-O_{12i}-1]+O_{12i}-1$

$B = [K2 * X_{1i}-O_{12i}-1]+O_{12i}-1$

$MACD_t = A-B$

$I_{8i} = \Sigma^{i}_{t=i-n} MACD_t$

$O_{18i} = W_8 * I_{8i}$

Step 2.9: Find lowest Price

$O_{19i} = Lowest (X_{1n}) * W_9$

Step 2.10: Find VAR

$I_{10i} = \Sigma^{i}_{t=i-n} (X_{1t}-\mu_n)^2$

$O_{110i} = W_{10} * I_{10i}$

Step 2.11: Find UPBANDS

$\mu = \left(\Sigma^{i}_{t=i-n} X_{1t}\right) / n$

$SD = SQRT \left(\Sigma^{i}_{t=i-n} (X_{1t}-\mu)^2 / n\right)$

$I_{11i} = \mu * 2SD$

$O_{111i} = W_{11} * I_{11i}$

Step 2.12: Find MOM

$I_{12i} = X_{1i}-X_{1i-n}$

$O_{112i} = W_{12} * I_{12i}$

Step 2.13: Find STOCH %K

$I_{13i} = [X_{1i}-Min (X_{4n})]/[Max (X_{3n})-Min (X_{4n})]$

$O_{113i} = W_{13} * I_{13i}$

Step 2.14: Find %R

$I_{14i} = [Max (X_{3n})-X_{1i}]/[Max (X_{3n})-Min (X_{4n})]$

$O_{114i} = W_{14} * I_{14i}$

Step 2.15: Find TR

$TH = Max (X_{3i}, X_{1i}-1)$

$TL = Min (X_{4i}, X_{1i}-1)$

$I_{15i} = TH-TL$

$O_{115i} = W_{15} * I_{15i}$

Step 2.16: Find AVG

$I_{16i} = X_{1i}+X_{2i}+X_{3i}+X_{4i}$

$O_{116i} = W_{16} * I_{16i}$

Step 2.17: Find MED

$I_{17i} = X_{3i}+X_{4i}$

$O_{117i} = W_{17} * I_{17i}$

Step 2.18: Find WCP

$I_{18i} = X_{3i}+X_{4i}+(2 * X_{1i})$

$O_{118i} = W_{18} * I_{18i}$

Step 2.19: Find SD

$I_{19i} = SQRT \left(\Sigma^{i}_{t=i-n} (X_{1t}-\mu)^2 / n\right)$

$$O_{119i} = W_{19} * I_{19i}$$

Step 2.20: Find FTS

$$Slop_i = \frac{\Sigma_{t=i-n}^{i} (X_{0t} - \mu X_{on}).(X_{1t} - \mu X_{1n})}{\Sigma_{t=i-n}^{i} (X_{ot} - \mu X_{on})^2}$$

$$I_{20i} = \mu_{X_{1n}} - slop * \mu_{X_{on}}$$

$$O_{120i} = W_{20} * I_{20i}$$

End loop

End

Layer 2; Standardization regression layer: The values in all features are standardized. The inputs of this layer merge the inputs and outputs of layer 1. Consequently, the number of output neurons is 27 according to the number of features with mean equal to zero and standard deviation equal to one.

The main reason of this process is to avoid the variation of large values, i.e., to prevent the learning deletion because these large values. The theoretical description Algorithm 3 shows the standardization layer.

Algorithm 3; Standardization regression layer:

Input: Array (Y_{vk}) marge two arrays (traditional features (X_{hk}) and array of TIs (O_{1jk}))

Output: array of output neurons (O_{2vk})

Begin

Step 1: The input neurons are formed in two dimensions according to the number of standard features plus TIs represent columns and input data represent rows. Additionally, the initial weights (T) are set to constant value equals to ($1/SD_V$)

Step 2: For g = 1 to v:

A) Compute Mean of each input neurons of g (μ_g)

B) Compute standard deviation of each input neurons of g (SD_g)

Step 3: For each output neuron:

A) Compute the weight of each neurons. It is computed as:

$T_{2g} = 1/SD_g$

B) The final value of output neuron is computed as:

$$O_{2gk} = T_{2g} * (Y_{gk} - \mu_g)$$

End

Layer 3; Regression predictor layer: The basic idea of this layer has developed prediction model through Optimize Multilayer Perceptron (OMLP) Nns with one hidden layer by minimizing the given loss function (squared error) using Quasi-Newton method based on projected gradients instead of using standard gradient descent.

The propose predictor model composes of three layers. Input is the first layer and it is compatible with input variables of a problem with a node for each input variable (27 neurons). The hidden layer is a second layer, which is used to capture non-linear relationships between variables. The third layer represents the output layer, which is used to provide the expected values.

A sigmoid (logistic) activation function is used in the hidden layer. Also, this activation functions is used in output layer. Two neurons are determined in hidden layer in addition to bias. The value of tolerance is equal to (1.0E-6). The weights are optimized by reducing the error value. Definitely, the error reduction contribute in improving the prediction accuracy. Optimization algorithm consists of many steps to improve the MLP as shows in Algorithm 4.

Algorithm 4; Optimization MLP using BFGS method:

Input: Array (Q_{vk}) = O_{2vk} (the output neurons values of layer 2)

Output: Array of updated weights

Begin

Definition symbols:

X: vector of all weight in NNs

SPDM: Symmetric Positive Definite Matrix

LT: Lower Triangle

DM: Diagonal Matrix

CF: Converges Factor

α: Step Length

Step 1: Initialization

Initial weights to vector X

Compute gradient to vector G using X

Compute matrix factorization of SPDM using LT and DM

Ridge = 0.01,

Step 2: For i = 0 to max-iterations Do

Step 3: compute the Jacobian gradient:

$G_i = G_i + Ridge * X_i$

Step 4: while CF≠0 Do

Step 5: Apply the Langrage multiplier to test the convergence

Step 6: Compute newton direction to vector D using inverse G

Step 7: Find the upper bound of X to determine the best value according to

$$X_{i+1} = X_i + \alpha D_i$$

Step 8: Update G depend on X, LT, DMA and LM until CF = 0 STOP

Step 9: Go to step 2

End

Dataset: The dataset to study is provided from Yahoo finance's stock historical data. The experimental tests with historical data set are taken from three global stock markets such as (S and P500, DOW and NASDAQ). The periods of S and P500 and Dow are selected from 18/10/2013 to 14/10/2016 while the period of NASDAQ is selected from 10/09/2013 to 06/09/2016. The historical data is collected in daily basis (open, high, low, close, volume, adj close). In total 754 of instances to those, three markets.

RESULTS AND DISCUSSION

Two metrics (RMSE and MAE) are used to evaluate the proposed regression predictors model. Additionally, a comparative study with two standard regression predictors KNN and SVM. The DNN-RP, KNN and SVM testing with cross-validation (10-fold) and compared for checking the difference from the prediction on same points of view. The results of RMSE and MAE to these regression predictors have applied on four datasets with all features are summarized in Table 3 and 4, respectively. The similar parameters with datasets contain only standard features are summarized in Table 5 and 6.

Table 3: RMSE with all features

All features	FDLNN-RP	K-NN	SVM
DOW	10.6064	74.4541	28.5042
S&P500	1.4663	7.8649	3.5967
Iraq	0.0051	0.0272	0.0095
NASDAQ	5.5079	27.6939	9.8766

Table 4: MAE with all features

All features	DNN-RP	K-NN	SVM
DOW	15.3809	100.2824	37.2175
S&P500	2.0292	11.4250	4.7702
Iraq	0.0070	0.0350	0.0128
NASDAQ	7.7800	35.6283	13.2411

Table 5: RMSE with standard features

Only S. Features	DNN-RP	K-NN	SVM
DOW	16.7513	47.6304	44.9398
S&P500	1.5607	7.2433	5.7856
Iraq	0.0127	0.0215	0.0189
NASDAQ	6.7506	19.9378	16.7909

Table 6: MAE with standard features

Only S. Features	DNN-RP	K-NN	SVM
DOW	12.1137	33.9036	33.2328
S&P500	1.1271	5.0226	4.2225
Iraq	0.0097	0.0150	0.0131
NASDAQ	4.8585	14.8683	12.5810

CONCLUSION

The DNN-RP is proposed as new model which involves an automatic and integrated process in addition to the optimization mathematical model to solve the convergence problem in the NN. The historical stocks data to generate decision procedures that give references to predict price in the SM. Then, technical indicators are demonstrated impact on the efficiency of the prediction model. The investors can be use a proposed model as useful tools to make the right decision on the market based on the analysis of the historical prices data of stocks and to find any predictive information from these data.The results for the proposed model are perfect after compared them with researcher prediction models (KNN and SVM). Also, benchmark of Iraq is proven accuracy when comparing with DOWJONES and S and P500 after implementing these three models.

REFERENCES

Atsalakis, G.S. and K.P. Valavanis, 2009. Surveying stock market forecasting techniques-Part II: Soft computing methods. Expert Syst. Applic., 36: 5932-5941.

Boonpeng, S. and P. Jeatrakul, 2016. Decision support system for investing in stock market by using OAA-neural network. Proceedings of the 8th International Conference on Advanced Computational Intelligence (ICACI), February 14-16, 2016, IEEE, Chiang Mai, Thailand, ISBN: 978-1- 4673-7780-5, pp: 1-6.

Chavan, P.S. and S.T. Patil, 2013. Parameters for stock market prediction. Intl. J. Comput. Technol. Appl., 4: 337-340.

Chen, X. and Z.J. He, 2015. Prediction of stock trading signal based on support vector machine. Proceedings of the 8th International Conference on Intelligent Computation Technology and Automation (ICICTA), June 14-15, 2015, IEEE, Nanchang, China, ISBN:978-1-4673-7644-0, pp: 651-654.

Dean, J., 2014. Big Data, Data Mining and Machine Learning: Value Creation for Business Leaders and Practitioners. John Wiley and Sons, Hoboken, New Jersey, USA., ISBN:9781502462916, Pages: 318.

Fernandez-Blanco, P., D.J. Bodas-Sagi, F.J. Soltero and J.I. Hidalgo, 2008. Technical market indicators optimization using evolutionary algorithms. Proceedings of the 10th Annual Conference Companion on Genetic and Evolutionary Computation, July 12-16, 2008, ACM, Atlanta, Georgia, USA., ISBN:978-1-60558-131-6, pp: 1851-1858.

Han, J. and M. Kamber, 2006. Data Mining Concepts and Techniques. 2nd Edn., Morgan Kaufmann, San Francisco, CA, USA.

Luo, B., Y. Chen and W. Jiang, 2016. Stock market forecasting algorithm based on improved neural network. Proceedings of the 8th International Conference on Measuring Technology and Mechatronics Automation (ICMTMA'16), March 11-12, 2016, IEEE, Macau, China, ISBN:978-1-5090-2311-0, pp: 628-631.

Mingyue, Q., L. Cheng and S. Yu, 2016. Application of the artifical neural network in predicting the direction of stock market index. Proceedings of the 10th International Conference on Complex, Intelligent and Software Intensive Systems (CISIS), July 6-8, 2016, IEEE, Fukuoka, Japan, ISBN:978-1-5090-0988-6, pp: 219-223.

Naeini, M.P., H. Taremian and H.B. Hashemi, 2010. Stock market value prediction using neural networks. Proceedings of the 2010 International Conference on Computer Information Systems and Industrial Management Applications (CISIM'10), October 8-10, 2010, IEEE, Krackow, Poland, ISBN:978-1-4244-7817-0, pp: 132-136.

Nemes, M.D. and A. Butoi, 2013. Data mining on Romanian stock market using neural networks for price prediction. Inf. Econ., 17: 125-136.

Pang-Ning, T., S. Michael and K. Vipin, 2005. Introduction to Data Mining. 1st Edn., Pearson, Addison Wesley, Boston, Massachusetts, USA., ISBN-13: 978-0321321367, Pages: 769.

Panigrahi, S.S. and J.K. Mantri, 2015. A text based decision tree model for stock market forecasting. Proceedings of the International Conference on Green Computing and Internet of Things (ICGCIoT), October 8-10, 2015, IEEE, Noida, India, ISBN:978-1-4673-7911-3, pp: 405-411.

Robert, D., J.M. Edwards and W.H.C. Bassetti, 2001. Technical Analysis of Stock Trends. 8th Edn., CRC Press, Boca Raton, Florida, USA., Pages: 711.

Shen, S., H. Jiang and T. Zhang, 2012. Stock market forecasting using machine learning algorithms. Master Thesis, Department of Electrical Engineering, Stanford University, Stanford, California, USA.

Yefimochkin, O., 2011. Fundamental: Using macroeconomic indicators and genetic algorithms in stock market forecasting. Master Thesis, Instituto Superior Tecnico, Lisbon, Portugal.

Yetis, Y., H. Kaplan and M. Jamshidi, 2014. Stock market prediction by using artificial neural network. Proceedings of the 2014 International Conference on World Automation Congress (WAC), August 3-7, 2014, IEEE, Waikoloa, Hawaii, USA., ISBN:978-1-8893-3549-0, pp: 718-722.