An Efficient System for Stock Market Prediction

Ashraf S. Hussein[1], Ibrahim M. Hamed[2], and Mohamed F. Tolba[3]

[1] Arab Open University, Headquarters, P.O. Box: 3322 Al-Safat 13033, Kuwait
ashrafh@acm.org
[2,3] Ain Shams University, Cairo, 11566, Egypt
{ibrahim.hamed,fahmytolba}@gmail.com

Abstract. This paper presents an efficient system for accurate, confident, general and responsive stock market prediction, employing Artificial Neural Networks (ANN). For technical indicators, Multi-Layer Perceptron (MLP) ANN is used and trained with Kullback Leibler Divergence (KLD) learning algorithm because it converges fast in addition to offering generalization in the learning mechanism. On the other hand, Radial Basis Function Neural Network (RBFNN) trained with Localized Generalization Error (L-GEM) is used for candlesticks patterns. The accuracy, generalization and statistical-significance of the developed system were confirmed through various local and international data sets. Next, sensitivity analysis was conducted for the different parameters that influence the system efficiency metrics. In order to have responsive prediction, the proposed system was evolved, employing concurrent programming to get benefit from the off-the-shelf multi-core architectures. Then, the performance of the developed system was evaluated to confirm acceptance scalability and utilization.

Keywords: Stock market prediction, technical indicator, candlesticks patterns, artificial neural networks, blind source separation, multi-core architecture, concurrency.

1 Introduction

Stock market prediction is one of the greatest challenges for experts and researchers who work in the financial sector. This topic has been tackled by many research groups, aiming at overcoming the accuracy, confidence and generalization challenges [1,2,3,4,5,6,7,8]. However, when the developed prediction models come to be applied, in the real stock markets, their results are not successful enough to consistently "beat the market", especially from accuracy and generalization points of view [5]. Ever since prediction was performed manually, technical experts could reach an accurate prediction pattern, but they used to miss the current transaction, either due to lack of timing or confidence [5], [9]. Therefore, accuracy is not the unique target, and considering generalization, prediction time and confidence along with accuracy of signals' prediction is very crucial [1], [5], [10]. To emphasize the effect of the prediction time, consider the example of a 5 hours trading session, with tick frequency of one tick per second. If the prediction system takes 1.5 seconds to provide the required

D. Filev et al. (eds.), *Intelligent Systems'2014*,
Advances in Intelligent Systems and Computing 323, DOI: 10.1007/978-3-319-11310-4_76

results, after 16 minutes the prediction comes 8 minutes late, and the signal might not only be delayed, but also it might be complementary to the current situation of the security. Also, the last 2 hours prediction results of the trading session will reach out after the session is being closed. In this way, considering the response time (prediction time) in addition to the accuracy and confidence has been pursued by some research groups [5]. High Performance Computing (HPC) techniques have been considered in order to have somewhat "real-time" predictions [11], but the specialized HPC computational resources (such as Computational Grids) are fairly sophisticated and not available for wide range of users and financial experts.

In this paper, an efficient system for stock market prediction is proposed to overcome the existing challenges, trying to consider the prediction time as a primary crucial factor in addition to the accuracy, confidence and generalization of the stock market prediction.

2 Previous Work

The state-of-the-art stock market prediction models and techniques have been surveyed in [1], [3], [5], considering the accuracy, confidence and generalization issues. Optimizing the performance of such prediction models towards real-time predictions has attracted less research groups. Early trials to enhance the prediction performance were based on sentimental information (such as market news) [12] or new stock market data forms (such as candlestick and point and figures) [13]. Nguyen et al. [14] tried to optimize the performance of the MLP ANN, as its performance drops when the network size increases. They proposed a new technique based on Cyclic Self-Organizing Hierarchical Cerebella Model Arithmetic Controller (CSOHCMAC). This technique exhibited high efficiency in terms of accuracy and response time, but its major drawback was the large memory requirements; the ratio of memory consumption between regular MLP and CSOHCMAC was around 1:125. On the other hand, some of the researchers have focused on sentimental factors like processing the market news and generating "Buy" and "Sell" signals. This wave was started by Ahmad et al. [11], employing the Financial Information Grid (FinGrid). They proposed a distributed environment, using Globus and Java Commodity Grids, to offer services by working on both qualitative and quantitative market data, as they added text analysis to the market news, along with the standard technical analysis indicators. This way the market sentiments were determined using the text analysis. Huang et al. [15] proposed a system for financial news headline agent to support the investors through the "Buy" and "Sell" decision making in the stock market of Taiwan. It receives the real-time market news headlines, published by the leading electronic newspapers in Taiwan, and employs optimized text mining techniques along with weighted association rules to predict the fluctuation in the Taiwan Stock Exchange Financial Price Index of the next trading day. The experimental results revealed that this system achieved significant performance.

Other research work is concerned with modern charting techniques for stock market prediction. Fu et al. [16] used both the rule-based and template-based approaches for stock charts pattern detection, relying on the Perceptually Important Points (PIPs).

As a result of this study, the authors recommended a hybrid model that integrates both of the template-based and rule-based approaches, employing the advantages of each. Li et al. [17] proposed RBFNN trained with L-GEM for candlesticks pattern detection of the morning star pattern only. This study avoided up to 69% of false patterns on the Shenzhen stock market. Following the recommendations of [17], Xiao et al. [6] applied four RBFNNs trained by localized L-GEM method, each of them corresponds to a particular candlestick pattern. Their strategy was found to be responsive with less accuracy. In the same context, Jasemi et al. [18] presented a model for stock market based on a supervised Feed-Forward ANN and technical analysis of the Japanese Candlesticks. They used ANN as a regression model to produce key parameters, from their independent variables, for technical analysis pattern detection. A raw-data based approach and signal based approach were used for defining the independent variables. The empirical results of these two approaches exhibited acceptable prediction for triggering "Buy" and "Sell" signals.

The aforesaid techniques and systems are quite promising, but they still experience problems related to accuracy, confidence degree or generalization, especially when trying to have "real-time" predictions. The issues of accuracy, confidence degree and generalization have been extensively tackled in Hamed et al. [3] to have a general model that can predict securities from different sectors and stock markets. This model is capable of adapting nonlinearities in the stock market and the un-correlated data of the different securities in various stock markets. The proposed technique adopts MLP ANN and KLD learning algorithm to enhance the performance of the proposed ANN. KLD, being a blind source separation technique, helps in solving the generalization issue of the prediction problem, keeping the model accuracy. The accuracy and generalization of the proposed prediction model were validated through wide range of stock markets, including the Microsoft stock, from wall-street market, and various data sets, from different sectors of the Egyptian stock market. In addition, the statistical-significance of the prediction results was confirmed through standard ANOVA test [3].

In this paper, the proposed system, developed based on the prediction model of [3], was evolved to consider the candlestick patterns' detection using RBFNN trained with L-GEM [17]. Then, the system was re-innovated and optimized; targeting the multi-core off-the-shelf architectures, using concurrent programming to reduce the computing time towards real-time predictions. Finally, the performance of the proposed system was evaluated via implementing the OKAZ's profile [19] based on both technical analysis and candlesticks to confirm accepted scalability and utilization. The proposed system exhibited responsive accurate results with high confidence degree and acceptable scalability level.

3 Prediction Model

The proposed model comprises of several stages as shown in Fig. 1. The first stage is concerned with the input selection. Next, the appropriate preprocessing is performed on the selected input data. Such preprocessing might be computing of indicators, fundamental assets evaluation or even data classification for the supervised learning of the ANN. The data is then passed to the ANN to be trained for the classification purposes. The main objective of the learning algorithm is to update the weights between the ANN neurons in order to minimize the error in the prediction results.

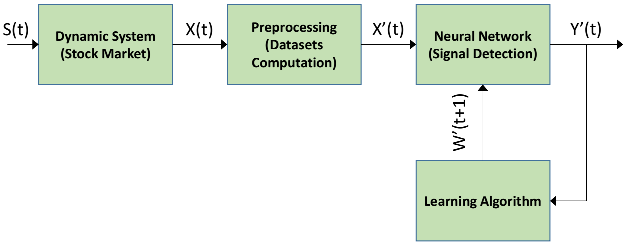

S(t)
Dynamic System (Stock Market)
X(t)
Preprocessing (Datasets Computation)
X'(t)
Neural Network (Signal Detection)
Y'(t)
W'(t+1)
Learning Algorithm

Fig. 1. Block diagram of the proposed model

3.1 Working Data

Stock Market's Daily Data

Daily data is the data used in the stock market daily transactions, and it may be called market summary. The daily data is composed of (a) open price, which is the first trading price for a security on a given trading session, (b) close price, which is the final trading price for a security on a given trading session, (c) high price, which is the highest trading price for a security during a given trading session, (d) low price, which is the lowest trading price for a security during a given trading session, (e) volume, which is the amount of trades transacted for a security on a given trading session.

Candlesticks Charts

Japanese candlesticks' charting was first used around 1850. According to Nison [20], it was first developed in Sakata Tow by Homma, a legendary rice trader. It was later modified over years to form the one currently used in the stock market of today. Japanese candlesticks are preferred by most of the technical analysts, since it provides more visual information in the form of well-known set of patterns. A white Marubozu is formed when the low price equals the opening price and the high price equals the closing price. This pattern reflects that buyers controlled the price action during the whole candle period (day, hour, etc.), according to graph aggregation [21].

3.2 Preprocessing

The input to this stage is the stock market daily data for a chosen security. The preprocessing step is concerned with computing the technical indicators, which is considered to be an input to the ANN.

3.3 Artificial Neural Network (ANN)

For the technical indicators, the MLP ANN architecture has one hidden layer. Sigmoid function with range [-1, +1] is used, as the activation function for each neuron. The number of input neurons is equal to the number of variables in the data set while the number of hidden neurons equals to twice the input neurons. This was found to perform better after several trials of different hidden neurons combination. The signal is being classified into three classes “Buy”, “Sell” or “Hold”. So, three output neurons are being used in the output layer. Each neuron should have the value [0 – 1] indicating the class it belongs to. For a given run, the output (0.8 0.12 0.08) means it is a “Buy” signal, since it is closer to the buy class, while (0.05 0.85 0.1) is a “Sell” signal. For any output to be valid, it should belong only to one class [3]. For candlestick patterns, RBFNN ANN is adopted [17].

3.4 Learning Algorithm

Through the iterative supervised learning of the MLP ANN, the data is processed through an intermediate stage to normalize the output of the current stage before entering the next iteration. Due to the nature of the KLD, the input to this function must be in the form of a probability distribution function, i.e. the magnitude of the output vector equals one. Therefore, the output vector is normalized to match this criterion. Then, it is passed to the KLD to compute the divergence between the output signal and the desired signal. The weights are updated according to [3]. For RBFNN, the ANN is trained using L-GEM algorithm [17].

4 System Description

4.1 System Architecture

The proposed system was designed to utilize multi-core architectures via shared memory model. The system consists of a data source, a processing unit, which includes a parallelization root and *n* processing cores, and output prediction results as illustrated in Fig. 2. The data source is the stream of live data feed that is entered to the system. The parallelization root (manager) is responsible for loading the data to the shared memory, distributing the work over the processing cores (workers), performing load balancing and consolidating the results from various cores to have the final prediction result. The processing core *i* is responsible for performing a prediction task as scheduled by the parallelization root. The shared memory model, adopted in this system, aims at minimizing the intercommunication among cores.

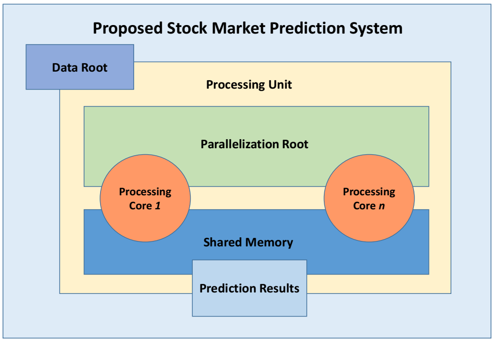


Fig. 2. The proposed system components

4.2 Data Flow

Initially, data is obtained from the live feed and passed to the parallelization root. Then, it divides the work load to *n* tasks, according to the selected profile or group of signal detection tasks, and sends the work to the available processing cores. The root applies the first free node selection mechanism for workload distribution among the processing cores. In this way, each processing unit performs the required data preprocessing, applies the prediction model then sends the results back through the shared memory. The parallelization root, in turn, consolidates the results from each processing core and generates the final prediction results, as shown in Fig. 3.

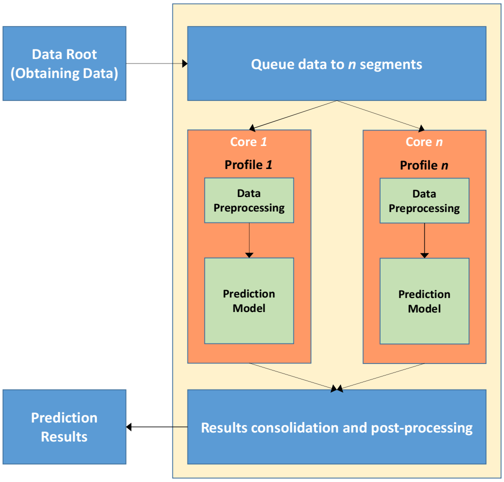


Fig. 3. Main data flow diagram

4.3 Implementation

The prediction models are based on MLP ANN trained with KLD [3] for the technical indicators and RBFNN trained with L-GEM [17] for the candlesticks pattern detection.

The considered set of indicators, in this implementation, includes: Simple Moving Average (SMA) [3], Exponential Moving Average (EMA) [3], Average True Range (ATR) [22], Stochastic Oscillator (SO) [23], Moving Average Convergence Divergence (MACD) [24], Average Directional Index (ADX) [22] and Candlestick Patterns [20], [21].

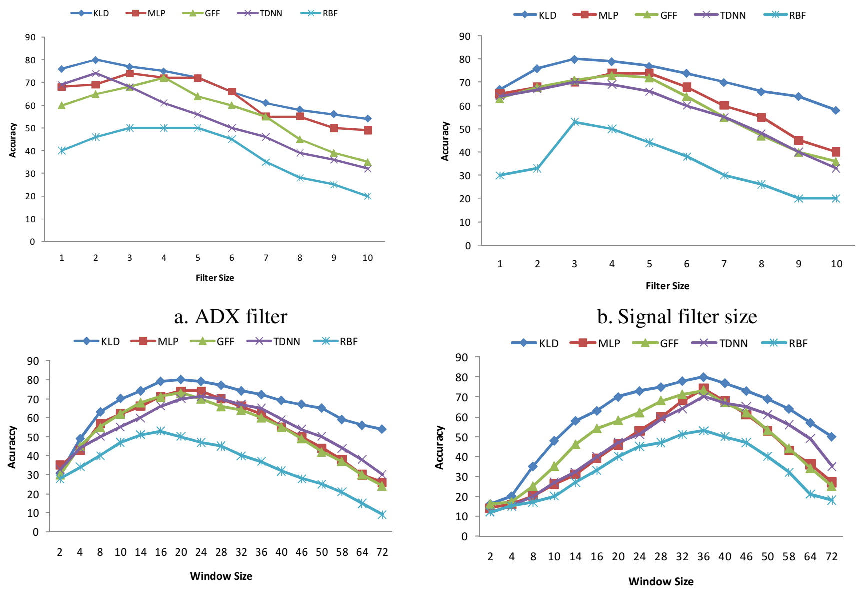
The input is used to calculate the indicators mentioned above with a given window and filter size. The selection of the appropriate window and filter sizes is based on a sensitivity analysis to obtain the "optimum" value of the aforesaid sizes. After indicators are being calculated, they all proceeded to a rule based system. This rule based system classifies the input signals into "Sell", "Hold" or "Buy". These classifications are used later in the learning stage.

5 Results and Discussions

In this research work, the proposed model was validated using the daily data from the Egyptian local stock market, including EFG Hermes Holding (HRHO), El Nasr Clothing - Textiles Co. (KABO), and Egyptian Electrical Cables (ELEC) with a total of 1900 records each. In order to consider the mature international stock markets in our validation, Microsoft (MS) stock was also encountered, from March 1999 to August 2008, with a total of 2600 records [19]. A sample 15% of this data was selected randomly for testing purposes. The selected data sets were meant to cover a wide spectrum of the stock market in terms of sectors, currency, trading volume and session type. First, HRHO was selected, from the investment sector, as it has a very active security with a large trading volume and is traded with the local currency. KABO was selected from the industrial sector, which has large trading volumes and is traded in USD. ELEC was selected from the Off-Trading Session (OTS), which is a 30 min session by the end of the trading day for corporates, which have some financial or legal violations and have a small trading volume.

Sensitivity analysis was carried out in order to identify the appropriate window size for every selected indicator in the data sets under consideration. Beside the window size of the indicators, there are two other filters. The first one is for the ADX signal, and it aims at ensuring the strength of the trend, i.e. to remove noise and false sudden moves in the price. The second one is for the "Buy", "Sell" or "Hold" signals. The purpose of this filter is to validate the signal and remove noise due to spikes in the price movement. Since rumors might influence security price movement, especially in growing markets like Egypt, this generates noise that lasts over a relatively short period (one to two trading sessions). Afterward, the signal is corrected again to match the actual value of the assets represented by the security. Sensitivity analysis was carried out for four variables: SMA window size, EMA window size, ADX filter and signal filter. For the window size, the size range (2 to 72) was considered while the range (1 to10) was tried for the filters. For each variable, a value, from its given range, is tried with all possible combination of the other 3 variables. Variable ranges could not be larger because this leads to short term and medium term signals, and also trends and movements disappear. Consequently, the prediction model accuracy will decrease because the ANN will capture only long term trades and will fail to detect short term and medium term trades.

Results from the proposed model were compared to that of the efficient ANN architectures mentioned in the research work of [7], [25], [26], [27], after developing the corresponding prediction models (for comparison purposes). The sensitivity analysis, considering the four variables mentioned above, shows that the proposed model outperforms the other techniques while the model of [7] exhibited the worst accuracy. In addition, the optimum values for these variables were identified for the data sets under consideration. Fig. 4 shows sample of the conducted sensitivity analysis for HRHO data set.



a. ADX filter

b. Signal filter size

c. SMA window size

d. EMA window size

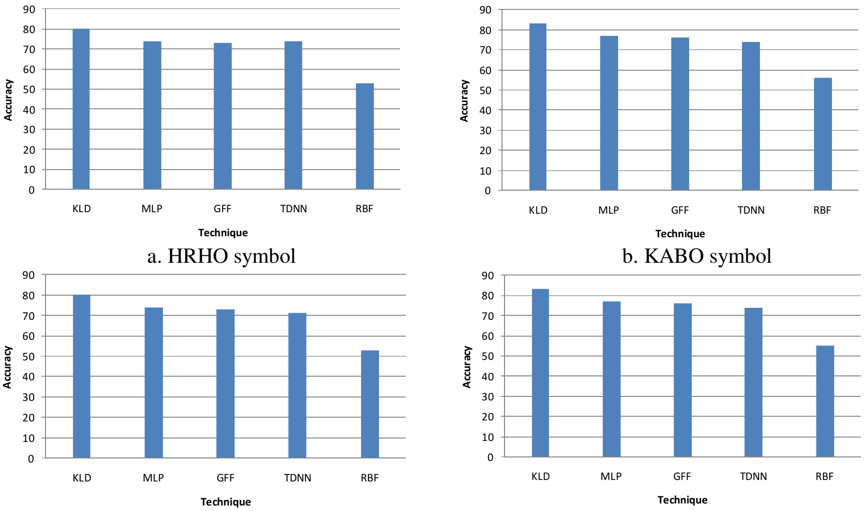
Fig. 4. Sensitivity analysis for the prediction model variables, HRHO data set

The results of the proposed model with the optimum values identified above, for the data sets under consideration, were then compared to that of the prediction models [7] [25] [26] [27] to confirm the accuracy of the proposed technique. The results were computed as an average of 20 separate cycles of training and testing. The maximum accuracy achieved by the proposed model is 83% as shown in Fig. 5.

ANOVA test [28] was carried out to ensure the statistical significance, i.e. the classification accuracy was not a result of random act. As mentioned earlier, the data sets were run for 20 complete cycles and the average accuracy results were used. ANOVA test resulted that the F value was 9.6 while the critical F value was 2.7 as shown in Table 1. Therefore, the means are significantly different and the generalization effect is real.

The performance of the proposed model was compared to that of the prediction models [7], [25], [26], [27]. The proposed technique converges faster than the other techniques, on the average of 3000 epochs, employing the Mean Squared Error as the error measurement function, as shown in Fig. 6. The other techniques always reach the maximum epochs and do not stop at the minimum error criteria. The number of epochs is set so that the neural network does not fall in the over fitting problem.

In order to examine the performance of the multi-core implementation of the developed system, OKAZ profile [19] was used for this purpose. After manipulating this profile, the confidence of the final result was calculated from their aggregation. Each



a. HRHO symbol

b. KABO symbol

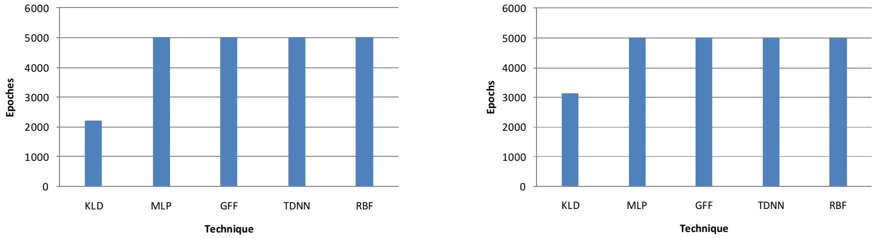
c. ELEC symbol

d. MS symbol

Fig. 5. Comparison between the accuracy of the proposed model and that of [7], [25], [26], [27]

Table 1. ANOVA test results

Source of Variation	SS	Df	MS	F	P-value	F_{crit}
Between Groups	151.4	3	50.45	9.8	1.5E-5	2.74
Within Groups	391.4	76	5.15			
Total	542.8	79				



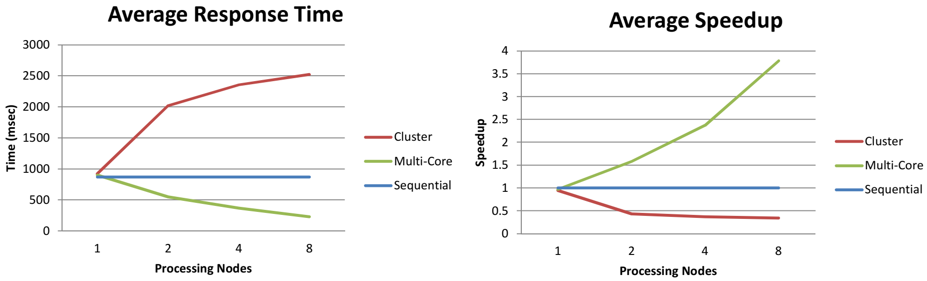
a. HRHO symbol

b. MS symbol

Fig. 6. Performance analysis of the proposed model

technique adds or removes a certain degree of confidence. The confidence percentage for OKAZ profile indicators are: ATR=20%, MA=5%, SO=5%, MCAD=10%, ADX=10% and Candle-Stick-Patters=40%. This provides a decision with confidence rate up to 90% as per the best prediction case.

This experiment was carried on Windows 2003 server with 8 cores. This server has 8 GB of shared RAM, and each processor is 1.6 GHz with turbo boost up to 2.0 GHz. In order to compare the multi-core implementation with the distributed memory one (cluster implementation), the Message Passing Interface (MPI) was used to develop the distributed memory version of the proposed system. This experiment was carried out on a cluster of workstations consists of 16 workstations. Each workstation has a 2.5 GHz Pentium Intel processor with dual-cores. The average of 10 different runs results were used for comparison purposes. As shown in Fig. 7.a, the response time of the multi-core implementation is less than that of the cluster one. This is originated from the added intercommunication latency among the cluster computing nodes. On the other hand, the multi-core implementation employs a shared memory model. Therefore, the intercommunication among cores is equivalent to the memory direct access operations. The speedup of the multi-core version outperforms the cluster version, and it achieved maximum speedup of 3.78 as shown in Fig. 7.b. For both versions, good processor utilization is achieved, within the range of (82%-94%). But, yet the multi-core has better utilization, as there are no bottlenecks in its workflow, as shown in Fig. 8. Generally, the system provided almost “real-time” results with good accuracy and generalization, with confidence degree up to 90%.



a. Average response time b. Average speedup

Fig. 7. Efficiency of the different implementations

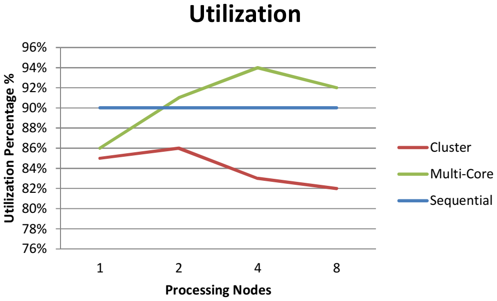


Fig. 8. Computing-node utilization for the different implementations

6 Conclusions

This paper proposes a new system for "real-time" stock market prediction with high accuracy, confidence rate and generalization. This system is based on the Artificial Neural Networks and employs off-the-shelf multi-core H/W architectures. For technical indicators, MLP ANN is used and trained with KLD learning algorithm because it converges fast and provides generalization in the learning mechanism. On the other hand, RBFNN trained with L-GEM is used for candlesticks pattern detection. The proposed system partitions the prediction mechanism to concurrent processes, and each process runs concurrently on a processing unit (core). Then, all the processes' results are consolidated by a unified parallelization root. The proposed system accuracy, confidence and generalization were confirmed through numerous data sets, covering wide sectors from the stock markets. In this way, three securities from the Egyptian local stock market in addition to MS security from a mature global stock market were addressed. Next, the OKAZ's profile based on both technical analysis and candlesticks was also considered for system performance analysis. The proposed system demonstrated its capability to provide real time accurate results with high confidence degree in addition to generalization. It reached more than 80% accuracy, maximum confidence degree of 90% and speedup of 3.78 with 8 processing units (cores). The results were confirmed to be statistically significant using standard ANOVA test. The future work includes optimizing the performance of the real-time predictions using GPU architectures to have better response time in addition to considering more market details.

References

1. Pimentel, M.A.F., Clifton, D.A., Clifton, L., Tarassenko, L.: Review: A Review of Novelty Detection. Journal of Signal Processing 99, 215–249 (2014)
2. Kristjanpoller, W., Fadic, A., Minutolo, M.C.: Volatility Forecast using Hybrid Neural Network Models. International Journal of Expert Systems with Applications 41(5), 2437–2442 (2014)
3. Hamed, I.M., Hussein, A.S., Tolba, M.F.: An Intelligent Model for Stock Market Prediction. International Journal Computational Intelligence Systems 5(4), 639–652 (2012)
4. Rout, M., Majhi, B., Mohapatra, U.M., Mahapatra, R.: Stock Indices Prediction using Radial Basis Function Neural Network. In: 3rd International Conference on Swarm, Evolutionary, and Memetic Computing, pp. 285–293 (2012)
5. Li, Y., Ma, W.: Applications of Artificial Neural Networks in Financial Economics: A Survey. In: 2010 International Symposium on Computational Intelligence and Design (ISCID 2010), vol. 10, pp. 211–214 (2010)
6. Xiao, W., Ng, W., Firth, M., Yeung, D.S., Cai, G.Y., Li, J.C., Sun, B.: L-GEM Based MCS Aided Candlestick Pattern Investment Strategy in the Shenzhen Stock Market. In: International Conference on Machine Learning and Cybernetics, vol. 1, pp. 243–248 (2009)
7. Quah, T.S.: Using Neural Network for DJIA Stock Selection. Engineering Letters 15(1), 126–133 (2007)

8. White, H.: Economic Prediction using Neural Networks: The Case of IBM Daily Stock Returns. In: IEEE International Conference on Neural Networks, vol. 2, pp. 451–458 (1988)
9. Lo, A.W., Mamaysky, H., Wang, J.: Foundations of Technical Analysis: Computational Algorithms, Statistical Inference, and Empirical Implementation. Journal of Finance 55(4), 1765–1770 (2000)
10. Murphy, J.: Technical Analysis of the Futures Markets: A Comprehensive Guide to Trading Methods and Applications. Prentice-Hal, New York (1986)
11. Ahmad, K., Taskaya-Temizel, T., Cheng, D., Gillam, L., Ahmad, S., Traboulsi, H., Nankervis, J.: Financial Information Grid –an ESRC e-Social Science Pilot. In: 3rd UK e-Science Programme All Hands Meeting, Nottingham, United Kingdom (2004)
12. Fung, P.C., Yu, X., Lam, W.: Stock Prediction: Integrating Text Mining. In: IEEE International Conference on Computational Intelligence for Financial Engineering, pp. 395–402 (2003)
13. Hwang, H., Oh, J.: Fuzzy Models for Predicting Time Series Stock Price Index. International Journal of Control, Automation and Systems 8(3), 702–706 (2010)
14. Nguyen, M.N., Omkar, U., Shi, D., Hayfron-Acquah, J.B.: Stock Market Price Prediction using Cyclic Self-Organizing Hierarchical CMAC. In: 9th International Conference on Control, Automation, Robotics and Vision, pp. 1–6 (2006)
15. Huang, C., Liao, J., Yang, D., Chang, T., Luo, Y.: Realization of a News Dissemination Agent Based on Weighted Association Rules and Text Mining Techniques. Expert Systems with Applications 37(9), 6409–6413 (2010)
16. Fu, T., Chung, F., Luk, R., Ng, C.: Stock Time Series Pattern Matching: Template-based vs. Rule-based Approaches. Engineering Applications of Artificial Intelligence 20(3), 347–364 (2007)
17. Li, H., Ng, W.W.Y., Lee, J.W.T., Binbin, S., Yeung, D.S.: Quantitative Study on Candle Stick Pattern for Shenzhen Stock Market. In: IEEE International Conference on Systems, Man and Cybernetics, SMC 2008, pp. 54–59 (2008)
18. Jasemi, M., Kimiagari, A.M., Memariani, A.: A Modern Neural Network Model to Do Stock Market Timing on the Basis of the Ancient Investment Technique of Japanese Candlestick. Expert Systems with Applications 38(4), 3884–3890 (2011)
19. Okaz (2014) `https://www.okazinvest.com/`
20. Nison, S.: Japanese Candlestick Charting Techniques, 2nd edn. Prentice Hall Press (2001)
21. Bigalow, S.: High Profit Candlestick Patterns. Profit Publishing LLC (2005)
22. Wilder, J.W.: New Concepts in Technical Trading Systems, 1st edn. Trend Research, Greensboro (1978)
23. Person, J.L.: A Complete Guide to Technical Trading Tactics: How to Profit using Pivot Points, Candlesticks & other Indicators, pp. 144–145. Wiley, Hoboken (2004)
24. Appel, G.: Technical Analysis Power Tools for Active Investors, p. 166. Financial Times Prentice Hall (1999)
25. Egeli, B., Ozturan, M., Badur, B.: Stock Market Prediction using Artificial Neural Net-Works. In: International Conference on Business, Hawaii (2003)
26. Jang, J.S.: ANFIS: Adaptive-Network-Based Fuzzy Inference System. IEEE Transactions on Systems, Man and Cybernetics 23(3), 665–685 (1993)
27. Grosan, C., Abraham, A., Ramos, V., Han, S.Y.: Stock Market Prediction using Multi Expression Programming. In: Portuguese Conference on Artificial intelligence, pp. 73–78 (2005)
28. Johnson, R.A., Wichern, D.W.: Applied Multivariate Statistical Analysis, 5th edn. Prentice Hall Upper Saddle River, NJ (2002)