Forecasting FTSE Bursa Malaysia KLCI Trend with Hybrid Particle Swarm Optimization and Support Vector Machine Technique

Lee Zhong Zhen[1], Yun-Huoy Choo[2],
Azah Kamilah Muda[3],

Faculty of Information and Communication Technology,
Universiti Teknikal Malaysia Melaka (UTeM),
Durian Tunggal, Melaka, Malaysia.
[1]james890526@yahoo.co.uk, [2]huoy@utem.edu.my,
[3]azah@utem.edu.my

Ajith Abraham[1,2]

[1] IT4Innovations, VSB-Technical University of Ostrava,
Czech Republic

[2] Machine Intelligence Research Labs (MIR Labs), USA
ajith.abraham@ieee.org

Abstract—**Stock trend forecasting is one of the important issues in stock market research. However, forecasting stock trend remains a challenge because of its irregular characteristic in the stock indices distribution, which changes over time. Support Vector Machine (SVM) produces a fairly good result in stock trend forecasting, but the performance of SVM can be affected by the high dimensional input features and noisy data. This paper hybridizes the Particle Swarm Optimization (PSO) algorithm to generate the optimum features set prior to facilitate SVM learning. The SVM algorithm uses the Radial Basis Function (RBF) kernel function and optimization of the gamma and large margin parameters are done using the PSO algorithm. The proposed algorithm was tested on a pre-sampled 17 years record of daily Kuala Lumpur Composite Index (KLCI) data. The PSOSVM approach is applied to eliminate unnecessary or insignificant features, and effectively determine the parameter values, in turn improving the overall prediction results. The optimized feature space of technical indicators of the algorithm is proven by the experimental results showing that PSOSVM has outperformed SVM technique significantly.**

Keywords-Stock Trend Forecasting; Particle Swarm Optimization; Support Vector Machine; RBF Kernal Function

I. INTRODUCTION

Stock trend forecasting is one of the important issues in stock market research due to its challenging characteristic. Various studies on the stock market forecasting have been researched and published [1-7]. Support Vector Machine (SVM) is able to generate good results [8] but are not robust in data with high dimensional input features and noisy. However, too many input features with noisy data may also affect the accuracy of the SVM algorithm [9]. Hence, a hybrid Particle Swarm Optimization – Support Vector Machine (PSOSVM) algorithm is proposed to overcome this problem so that more accurate predictions results can be produced. Particle Swarm Optimization (PSO) is an optimization algorithm, thus it is possible to select the optimal features for stock trend forecasting due to its simplicity and computational efficiency [49-53]. The main focus is to select the better features or attributes in stock trend forecasting so that these selected features can lead to a better classification result for SVM.

In this paper, we presented the implementation of a hybrid PSO and SVM technique to predict the 17 years Malaysia stock trend data. The proposed technique is implemented in MATLAB using the modified LIBSVM. The rest of the paper is organized as follows: Section 1 presents the introduction of the problem; Section 2 describes some related literature work; Section 3 discussed on the proposed PSOSVM technique; Section 4 depicts the experimental design, results and the discussion; and Section 5 draws the conclusion and several potential future directions to extend this research work.

II. RELATED WORK

Numerous studies and researches have been done to predict the future movement of stock market. Predicting the trend of the stock market is considered as a tough and challenging task in financial time-series forecasting. The main reason is because of the uncertainties involved in the movement of the stock market. Various factors affect the stock market and some of them include traders' expectations, events related to the government and economic conditions. Hence, predicting the movement of stock market price is quite a difficult job.

Over the years, many studies on the behavior of stock market prices have been published [10-14]. Stock index movement are not random but is highly stochastic, noisy, non-stationary and deterministically chaotic [15, 16]. Therefore, these stock indices usually behave in a highly non-linear and dynamic manner [17]. This has made the stock market prices prediction to be a very challenging work. However, the most important factor in gaining profit using financial prediction lies on the trend prediction but not the exact value of the future stock prices. The direction of the the stock index movement makes sufficiently significant influence to most of the investors to make decision as compared to the exact stock price. Thus, trend forecasting has emerged to be more practical than stock price prediction [8].

A number of artificial intelligence and machine learning techniques have been used for decades to predict the stock market trend. Although many Artificial Neural Networks (ANN) techniques have been widely used [18-21], Support Vector Machine (SVM) has gained its popularity over the

169

past few years. SVM is a non-parametric, non-linear classification technique, which has shown good results in many time series forecasting, i.e. the electric consumption forecasting, optical character recognition, medical diagnostics and many other fields [22]. Kim [23] uses the technical analysis indicators as data features to predict the stock market direction using SVM learning scheme. Reference [23] also concludes that SVM has outperformed the Back Propagation Neural Networks (BPNN) generally.

A. FTSE Burse Malaysia KLCI Index

Market indices are important tools when trying to anticipate future movement of a stock price. Major stock market dataset are commonly used by researchers in predicting the stock trend, i.e. the United States's stock indices including the Nasdaq index, the Dow Jones index and the S&P 500 index, the Nikkei index, Hang Seng index, etc [24-28]. However, other stock indices are equally important especially to the local community as they are better indicators to reflect the local economy status. Malaysian stock market refers to the Kuala Lumpur Composite Index (KLCI), which is now known as FTSE Bursa Malaysia KLCI. The KLCI's historical movements of the Malaysian stock market is updated in a 15 seconds frequency index calculation manner, tracking the market pulse closely and efficiently. It is a market indicator made up of primary market movers which will more suitably define market activities, thus serves as the representative of the country stock market.

B. Stock Trend Forecasting

Analysts from the early centuries denied the possibility of forecasting the stock market's future direction by giving various illustrations and example to disprove the existence of any forecasting techniques which are under research [29]. However, with the help of technical indicator, technical analysts have proven that there are actually many techniques which have been applied in the stock market field and these techniques have generated significant results which directly profits the stock market companies [16, 23, 30].

Recent research in using SVM to forecast the stock market has made significant progress. In 2001, Tay and Cao have used SVM for financial time series prediction [29, 31-33]. In 2003, Kim [23] has proposed to use technical indicators as input features for SVM to predict stock market direction. In the last few years, hybridization of AI techniques has also been proposed. In 2007, Kim and Shin [21] have implemented a genetic algorithm (GA) and neural network approach to improve the effectiveness of the ANN model by optimizing the number of time delays using GA. Besides that, Tu and Chung [34] have also proposed a hybrid of PSO and SVM in several classification problems to increase the classification accuracy by performing feature selection using PSO.

However, to the knowledge of the author, there is little published work focusing on the FTSE KLCI data [10-14, 28]. In 2005, Bahrun and Taib [28] used the Maybank and Tenaga stocks to predict stock trend using Artificial Neural Network (ANN). A combination of technical indicator and raw data was used in [10] to predict KLCI with ANFIS modeling in 2008. The fuzzy time series model was also used to forecast KLCI in 2011 [13].

C. Technical Indicator

Technical analysis indicators are derived from technical charts which help interpret the market trend. They are graphical or visual representations of the market activity in terms of upward or downward movements in stock prices over a period of time. Besides that, some technical analysis indicators also take into account trading volumes in the calculations. Leading indicators change before the underneath commodity change. The main benefits of leading indicators is the early signalling for entry and exit, it generate more signals and allow more opportunities to trade in trading markets. On the other hand, more signals and earlier signals mean that the changes of false signals will also increase that might increase the potential losses. Lagging indicators follow the price action and are also known as trend-following indicators. Trend-following indicators work best when markets or securities develop a strong trends and signal traders to take long position and keep them holding the position as long as the trend is intact. Trend indicators are not effective in trading or sideways markets and tend to lead to many false signals. Late signal that cause late entry is the only disadvantage of trend-following indicators.

A set of 25 technical indicators in [35] has been chosen for this research because they are commonly being used by financial experts. Some of the more important indicators include Williams %R, Momentum, Stochastic %K, Price Volume Trend and Rate of Change. However, traders must also take note of the strengths and weaknesses associated with most technical indicators. Hence, choosing the better technical indicators is important and they are those that have in the past, tested and proven to be successful.

III. THE PROPOSED PSOSVM TECHNIQUE

PSO is different from other evolutionary techniques [36] that it does not need evolution operation such as crossover and mutation. Moreover, information is shared among individuals along the searching process so that the entire population can be maintained, thus directing the search towards the best position in the search space. It has also fast converging characteristics and more global-searching ability at the beginning of the run and a local searching near the end.

Lin et al. [37] have used a hybrid approach of PSOSVM to measure the performance of the technique on a variety of problems. The following datasets are used: Australian, Boston housing, Breast cancer, Bupa live, German, Ionosphere, Pima, Sonar, Car Evaluation Database, Glass, Teaching Assistant Evaluation, Vehicle, Vowel and Wine; taken from the UCI machine learning repository [38] and Bioinformatic [39]. Results showed that a PSO-based approach is capable of searching for the optimal parameter values for SVM to obtain a subset of beneficial features. Besides that, Wu [40] has also applied the same technique

in short-term load forecasting to improve the security of the power system. Results showed that the forecasting method is effective and feasible in such time-series data.

In PSO, each single candidate solution is a particle in the search space. Each particle makes use of its individual memory and knowledge gained by the swarm as a whole to find the best solution [36, 41]. All of the particles have fitness values, which are evaluated by a fitness function (in this study is the SVM), and have velocities which directs the movement of the particles. During searching, each particle adjusts its position according to its own experience and the experience of its neighboring particle, and identifies the best position encountered by itself and its neighbors. The particles move through the problem space by following a current of optimum particles.

The architecture of the proposed PSOSVM technique is illustrated in Figure 1. In this technique, the prediction accuracy serves as the fitness function for the PSO method. In addition, PSO is also used to find the optimal parameters for SVM, i.e. gamma (γ) and C automatically. PSO performs as feature selection while SVM was used to predicts the results.
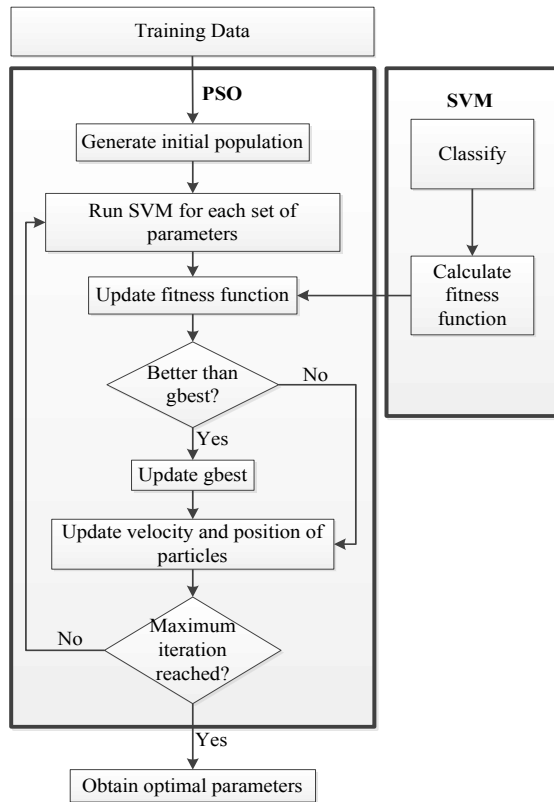

Figure 1. Flow Chart of PSOSVM.

The initial swarm is generally created in such a way that the population of the particles is distributed randomly over the search space. Each particle keeps track of its coordinates in the problem space, which are associated with the best solution the particle has achieved so far. Each particle is updated by following two factors, its own best previous experience (*pbest*) and the best experience among the population (*gbest*) in every iteration.

Based on the rules of particle swarm optimization, the required particle numbers are set, and then the initial coding alphabetic string for each particle is randomly produced. The adaptive functional values were data based on the particle features representing the feature dimension; this data was classified by a SVM to obtain classification accuracy; the SVM serves as an evaluator of the PSO fitness function. The fitness value for the SVM evolves according to the K-fold Cross-Validation. The data is separated into 10 parts and carried out training and testing a total of 10 times.

Each particle renewal is based on its adaptive value. The best adaptive value for each particle renewal is *pbest*, and the best adaptive value within a group of *pbest* is *gbest*. Once *pbest* and *gbest* are obtained, features of *pbest* and *gbest* particles are tracked with regard to their position and speed. A binary version of a PSO algorithm is used for particle swarm optimization [42]. The position of each particle is given in a binary string form that represents the feature selection situation. Each particle is updated according to the following equations.

$$v_{pd}^{new} = w \times v_{pd}^{old} + c_1 \times rand_1 \times \left(pbest_{pd} - x_{pd}^{old}\right) +$$

$$c_2 \times rand_2 \times \left(gbest_d - x_{pd}^{old}\right) \tag{1}$$

$$S\left(v_{pd}^{new}\right) = \frac{1}{1 + e^{-v_{pd}^{new}}} \tag{2}$$

If $\left(rand < S\left(v_{pd}^{new}\right)\right)$ then $x_{pd}^{new} = 1$; else $x_{pd}^{new} = 0$ (3)

The feature after renewal is calculated by the function $S\left(v_{pd}^{new}\right)$, in which the speed value is v_{pd}^{new}. If $S\left(v_{pd}^{new}\right)$ is larger than a randomly produced disorder number that is within (0, 1), then its position value $F_n, n = 1,2, \dots, m$ is represented as {1}. However, if $S\left(v_{pd}^{new}\right)$ is smaller than a randomly produced disorder number that is within {0~1}, then its position value $F_n, n = 1,2, \dots, m$ is represented as {0}.

A support vector machine (SVM) is a computer algorithm that learns by example to classify objects [43]. It is a new technique suitable for binary classification tasks, which is related to and contains elements of non-parametric applied statistics, neural networks and machine learning. In most cases where the data are not linearly separable such as in stock markets, the data would be consequently mapped to a higher-dimensional feature space. Therefore, if the data cannot be classified clearly in the current dimensional space, then the SVM will map them to a higher dimensional space for classification. Input data are mapped to a higher dimensional feature space by plotting a nonlinear curve.

Several kernel functions help the SVM in obtaining the optimal solution. The most frequently used kernel functions are the polynomial, sigmoid and Gaussian radial basis

function (RBF) [44-46]. Unlike a linear kernel function, the RBF kernel is widely accepted because it can classify multi-dimensional data. In addition, the RBF has fewer parameters settings than a polynomial kernel while maintain similar overall performance. Therefore, this research applies an RBF kernel function in the SVM to obtain optimal solution.

Two RBF parameters in SVM, C and γ must be set appropriately. The parameter C is a tuning parameter, which weights in-sample classification errors and thus controls the generalization ability of an SVM. The higher is C, the higher is the weight given to in-sample misclassifications, the lower is the generalization of the machine. In other words, higher C leads to higher specification of the machine. High specification means that the machine may work well on a training set but would perform miserably on a new sample. Bad generalisation may be a result of overfitting on the training sample. By choosing a low C, the risk of overfitting an SVM on the training sample is reduced. It can be demonstrated that C is linked to the width of the margin. The smaller is C, the wider is the margin, the more and larger in-sample classification errors are permitted. Parameter γ has a much greater influence on classification outcomes than C, because its value affects the partitioning outcome in the feature space. The higher is γ, the smoother is the threshold which separates the classes. However, an excessively large value for parameter γ results in overfitting [47]. In order to obtain the optimal values for the parameters, experiments on the dataset are needed since the values vary in every dataset. SVM offers a few advantages as below [48]: SVM provides a good out-of-sample generalization, if the parameters C and γ are appropriately chosen. SVM can be robust, even when the training sample has some bias. SVM delivers a unique solution, since the optimality problem is convex.

IV. THE EXPERIMENT SETTINGS

Both PSO and SVM are implemented in MATLAB. The SVM algorithm is modified from the LIBSVM tool to integrate it into MATLAB. The parameter values for the PSOSVM technique are set as in [36]. Both the cognition learning factor, c_1 and the social learning factor, c_2 are set to 2. Since the PSOSVM approach has a larger solution space, in terms of number of features, the number of solution evaluated is also larger. By setting the number of particles and generations to 8 and 250 respectively, the number of solution evaluated sums up to 2000. The searching range of parameter C of SVM is between 0.01 and 35000, while the searching range of parameter γ of SVM is between 0.0001 and 32 [36]. A paired sample t-test is also performed to compare the means of the two samples — SVM and PSOSVM using the SPSS 17 software.

V. DISCUSSIONS

A set of 15 technical indicator are selected by the proposed PSOSVM technique, i.e. Accumulation Distribution, Average True Range, Commodity Channel Index, Detrended Price Oscillator, Ease of Movement,

Money Flow Index, Performance, Price Volume Trend, Relative Strength index, SMA-10, Stochastic %K, TRIX, Typical Price, Volume Oscillator, and Weighted Close. PSOSVM has performed well even in situation of small training data and larger testing data. This is because the SVM implements the structural risk minimization principle and this leads to better generalization. The added advantage of providing a larger partition of training set to SVM is that it can generalize better and lead to better prediction accuracy for stock market price movement as shown in Figure 2.
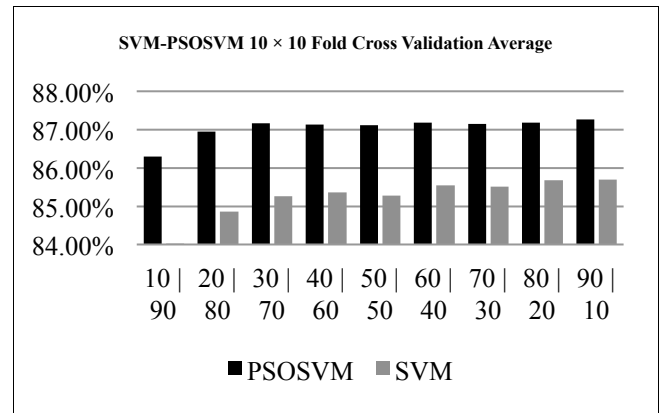

Figure 2. Classification Accuracy of PSOSVM vs SVM.

Comparison of the obtained results with SVM demonstrates that the proposed PSOSVM technique has better prediction accuracy. The PSOSVM approach is applied to eliminate unnecessary or insignificant features, and effectively determine the parameter values, in turn improving the overall prediction results. A common disadvantage of non-parametric techniques such as SVM is the lack of transparency of results. However, interpretation of results is possible and can rely on graphical visualization, as well as on a local linear approximation of the score. Results of this research were obtained with an SVM with RBF kernel function. SVM delivers a unique solution since the optimality problem is convex. This is an advantage compared to neural networks, which have multiple solutions associated with local minima and for this reason may not be robust over different samples.

Other kernel parameters for Particle Swarm Optimization (PSO) and Support Vector Machine (SVM) can be further optimized so that better generalization result can be produced. Since only the RBF kernel function is applied in this study, other kernel functions such as polynomial or sigmoid can be tested to verify their performances in compared to RBF. Technical indicators which serve as input features for the SVM can also be increased to forecast the stock market. Each technical analysis indicator has its own unique information. That is why it is important to be cautious of the strengths and weaknesses associated with each technical indicator. Many factors interact in the stock market including political events, general economic conditions and traders' expectations. Every field can be discovered to look for significant information to predict the stock market. Fundamental analysis can also be combined with technical

analysis to improve prediction accuracy. There are an infinite number of possibilities to improve stock trend forecasting accuracy. Hence, further research can be done to search for better approaches.

VI. CONCLUSION

This study presents a PSO-based approach, Particle Swarm Optimization–Support Vector Machine (PSOSVM) which is capable of searching for the optimal parameter values for SVM to obtain a subset of beneficial features. This optimal subset of features is then adopted in both training and testing to obtain the optimal outcomes in stock trend prediction. The proposed hybrid PSOSVM technique has been proven to be able to generate better results when compared to the Support Vector Machine (SVM) technique. With feature selection, the PSOSVM approach yielded a higher prediction accuracy rate. Since PSOSVM has proven to be better than the conventional SVM approach in prediction stock price movement.

REFERENCES

[1] H. Al-Qaheri, A. E. Hassanien, and A. Abraham, "Discovering stock price prediction rules using rough sets," vol. 18, pp. 181-198, 2008.

[2] Y. Chen, B. Yang, and A. Abraham, "Flexible neural trees ensemble for stock index modeling," Neurocomputing, vol. 70, pp. 697-703, 2007.

[3] A. Abraham, N. S. Philip, and P. Saratchandran, "Modeling Chaotic Behavior of Stock Indices Using Intelligent Paradigms," Journal of Neural, Parallel & Scientific Computations, vol. 11, pp. 143-160, 2003.

[4] C. Grosan and A. Abraham, "Stock market modeling using genetic programming ensembles," in Genetic Systems Programming, Nadia Nedjah et al. (Eds.), Springer Verlag Germany, 2006, pp. 131-146.

[5] Y. Chen, L. Peng, and A. Abraham, "Stock index modeling using hierarchical radial basis function networks," in 10th International Conference on Knowledge-Based & Intelligent Information & Engineering Systems, Germany, 2006, pp. 398-405.

[6] A. Abraham, C. Grosan, S. Y. Han, and A. Gelbukh, "Evolutionary multiobjective optimization approach for evolving ensemble of intelligent paradigms for stock market modeling," in MICAI 2005: Advances in Artificial Intelligence, ed: Springer, 2005, pp. 673-681.

[7] A. Abraham, B. Nath, and P. K. Mahanti, "Hybrid intelligent systems for stock market analysis," in Computational Science-ICCS 2001, Springer, 2001, pp. 337-345.

[8] N. Powell, S. Y. Foo, and M. Weatherspoon, "Supervised and Unsupervised Methods for Stock Trend Forecasting," in System Theory, 2008. SSST 2008. 40th Southeastern Symposium on, 2008, pp. 203-205.

[9] S. Pang, I. Havukkala, and N. Kasabov, "Two-class SVM trees (2-SVMT) for biomarker data analysis," in Advances in Neural Networks-ISNN 2006, Springer, 2006, pp. 629-634.

[10] Z. M. Yunos, S. M. Shamsuddin, and R. Sallehuddin, "Data Modeling for Kuala Lumpur Composite Index with ANFIS," in Modeling & Simulation, 2008. AICMS 08. Second Asia International Conference on, 2008, pp. 609-614.

[11] K. H. Lye, T. S. Yean, and K. L. Ming, "Modified binomial tree and market efficiency: the case for KLCI and LTCM," in Fuzzy Systems and Knowledge Discovery, 2009. FSKD'09. Sixth International Conference on, 2009, pp. 538-542.

[12] M. A. Maasar, Z. M. Rodzi, N. A. M. Nordin, and N. H. Kamis, "Portfolio insurance of a portfolio reflected by FTSE Bursa Malaysia KLCI," in Humanities, Science and Engineering (CHUSER), 2012 IEEE Colloquium on, 2012, pp. 513-517.

[13] L. Abdullah and C. Y. Ling, "A fuzzy time series model for Kuala Lumpur Composite Index forecasting," in Modeling, Simulation and Applied Optimization (ICMSAO), 2011 4th International Conference on, 2011, pp. 1-5.

[14] A. Omar and M. M. Jaffar, "Comparative analysis of Geometric Brownian motion model in forecasting FBMHS and FBMKLCI index in Bursa Malaysia," in Business, Engineering and Industrial Applications (ISBEIA), 2011 IEEE Symposium on, 2011, pp. 157-161.

[15] E. F. Fama, "The behavior of stock-market prices," The journal of Business, vol. 38, pp. 34-105, 1965.

[16] L. Yu, S. Wang, and K. K. Lai, "An online learning algorithm with adaptive forgetting factors for feedforward neural networks in financial time series forecasting," Nonlinear Dynamics and Systems Theory, vol. 7, pp. 97-112, 2007.

[17] X. Sui, Q. Hu, D. Yu, Z. Xie, and Z. Qi, "A Hybrid Method for Forecasting Stock Market Trend Using Soft-Thresholding De-noise Model and SVM," in Rough Sets, Fuzzy Sets, Data Mining and Granular Computing, ed: Springer, 2007, pp. 387-394.

[18] E. W. Saad, D. V. Prokhorov, and D. C. Wunsch, "Advanced neural network training methods for low false alarm stock trend prediction," in Neural Networks, 1996., IEEE International Conference on, 1996, pp. 2021-2026.

[19] J. A. Tome and J. P. Carvalho, "Market index prediction using fuzzy boolean nets," in Hybrid Intelligent Systems, 2005. HIS'05. Fifth International Conference on, 2005, p. 5 pp.

[20] Q. Ye, B. Liang, and Y. Li, "Amnestic neural network for classification: application on stock trend prediction," in Services Systems and Services Management, 2005. Proceedings of ICSSSM'05. 2005 International Conference on, 2005, pp. 1031-1034.

[21] H.-J. Kim and K.-S. Shin, "A hybrid approach based on neural networks and genetic algorithms for detecting temporal patterns in stock markets," Applied Soft Computing, vol. 7, pp. 569-576, 2007.

[22] N. Sapankevych and R. Sankar, "Time series prediction using support vector machines: a survey," Computational Intelligence Magazine, IEEE, vol. 4, pp. 24-38, 2009.

[23] K.-J. Kim, "Financial time series forecasting using support vector machines," Neurocomputing, vol. 55, pp. 307-319, 2003.

[24] K. Chaereonkithuttakorn, "The Relationship between the Stock Exchange of Thailand Index and the Stock Indexes in the United States of America," Master Thesis in Economics, Chiang Mai University, Chiang Mai, Thailand, 2005.

[25] S. Rimcharoen, D. Sutivong, and P. Chongstitvatana, "Prediction of the stock exchange of Thailand using adaptive evolution strategies," in Tools with Artificial Intelligence, 2005. ICTAI 05. 17th IEEE International Conference on, 2005, pp. 5 pp.-236.

[26] C. Worasucheep, "A New self adaptive differential evolution: Its application in forecasting the Index of stock exchange of Thailand," in Evolutionary Computation, 2007. CEC 2007. IEEE Congress on, 2007, pp. 1918-1925.

[27] S. Chaigusin, C. Chirathamjaree, and J. Clayden, "The use of neural networks in the prediction of the stock exchange of Thailand (SET) Index," in Computational Intelligence for Modelling Control & Automation, 2008 International Conference on, 2008, pp. 670-673.

[28] P. N. Bahrun and M. N. Taib, "Selected Malaysia stock predictions using artificial neural network," in Signal Processing & Its Applications, 2009. CSPA 2009. 5th International Colloquium on, 2009, pp. 428-431.

[29] B. G. Malkiel, A random walk down Wall Street: the time-tested strategy for successful investing: WW Norton & Company, 2003.

[30] W.-H. Chen, J.-Y. Shih, and S. Wu, "Comparison of support-vector machines and back propagation neural networks in forecasting the six major Asian stock markets," International Journal of Electronic Finance, vol. 1, pp. 49-67, 2006.

[31] L. Cao and F. E. Tay, "Financial forecasting using support vector machines," Neural Computing & Applications, vol. 10, pp. 184-192, 2001.

[32] F. E. Tay and L. Cao, "Application of support vector machines in financial time series forecasting," Omega, vol. 29, pp. 309-317, 2001.

[33] F. E. H. Tay and L. J. Cao, "Improved financial time series forecasting by combining support vector machines with self-organizing feature map," Intelligent Data Analysis, vol. 5, pp. 339-354, 2001.

[34] C.-J. Tu and L.-Y. Chuang, "Feature selection using PSO-SVM." IAENG International Journal of Computer Science, 33:1, 2007.

[35] P. J. Kaufman, Trading systems and methods: Wiley.com, 2013.

[36] Q. Wu, "Hybrid forecasting model based on support vector machine and particle swarm optimization with adaptive and Cauchy mutation," Expert systems with applications, vol. 38, pp. 9070-9075, 2011.

[37] S.-W. Lin, K.-C. Ying, S.-C. Chen, and Z.-J. Lee, "Particle swarm optimization for parameter determination and feature selection of support vector machines," Expert Systems with Applications, vol. 35, pp. 1817-1824, 2008.

[38] K. Bache and M. Lichman, "UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]," Irvine, CA: University of California, School of Information and Computer Sciences, 2013.

[39] C. Hsu, C. Chang, and C. Lin, "A practical guide to support vector classification. Department of Computer Science and Information Engineering, National Taiwan University, Taiwan," Taipei, Taiwan, 2003.

[40] Q. Wu, "A hybrid-forecasting model based on Gaussian support vector machine and chaotic particle swarm optimization," Expert Systems with Applications, vol. 37, pp. 2388-2394, 2010.

[41] G. Venter and J. Sobieszczanski-Sobieski, "Particle swarm optimization," AIAA journal, vol. 41, pp. 1583-1589, 2003.

[42] J. Kennedy and R. C. Eberhart, "A discrete binary version of the particle swarm algorithm," in Systems, Man, and Cybernetics, 1997. Computational Cybernetics and Simulation., 1997 IEEE International Conference on, 1997, pp. 4104-4108.

[43] B. E. Boser, I. M. Guyon, and V. N. Vapnik, "A training algorithm for optimal margin classifiers," in Proceedings of the fifth annual workshop on Computational learning theory, 1992, pp. 144-152.

[44] Y. Liao, S.-C. Fang, and H. LW Nuttle, "A neural network model with bounded-weights for pattern classification," Computers & Operations Research, vol. 31, pp. 1411-1426, 2004.

[45] H.-T. Lin and C.-J. Lin, "A study on sigmoid kernels for SVM and the training of non-PSD kernels by SMO-type methods," submitted to Neural Computation, pp. 1-32, 2003.

[46] K.-R. Muller, S. Mika, G. Ratsch, K. Tsuda, and B. Scholkopf, "An introduction to kernel-based learning algorithms," Neural Networks, IEEE Transactions on, vol. 12, pp. 181-201, 2001.

[47] M. Pardo and G. Sberveglieri, "Classification of electronic nose data with support vector machines," Sensors and Actuators B: Chemical, vol. 107, pp. 730-737, 2005.

[48] L. Auria and R. A. Moro, "Support Vector Machines as a Technique for Solvency Analysis," DIW Berlin, German Institute for Economic Research, No. 811, 2008.

[49] H. Izakian, A. Abraham, V. Snasel, Comparison of Heuristics for Scheduling Independent Tasks on Heterogeneous Distributed Environments, The 2009 IEEE International Workshop on HPC and Grid Applications (IWHGA2009), China, IEEE Press, USA, ISBN 978-0-7695-3605-7, pp. 8-12, 2009.

[50] H. Liu, A. Abraham and M. Clerc, Chaotic Dynamic Characteristics in Swarm Intelligence, Applied Soft Computing Journal, Elsevier Science, Volume 7, Issue 3, pp. 1019-1026, 2007.

[51] A. Abraham, Intelligent Systems: Architectures and Perspectives, Recent Advances in Intelligent Paradigms and Applications, Abraham A., Jain L. and Kacprzyk J. (Eds.), Studies in Fuzziness and Soft Computing, Springer Verlag Germany, ISBN 3790815381, Chapter 1, pp. 1-35, 2002.

[52] R. Thangaraj, M. Pant, A. Abraham and P. Bouvry, Particle Swarm Optimization: Hybridization Perspectives and Experimental Illustrations, Applied Maths and Computation, Elsevier Science, Netherlands, Volume 217, No. 1, pp. 5208-5226, 2011.

[53] H. Liu, A. Abraham, O.K. Choi and S.H.Moon, Variable Neighborhood Particle Swarm Optimization for Multi-objective Flexible Job-shop Scheduling Problems, The Sixth International Conference on Simulated Evolution And Learning (SEAL06), China, Springer Verlag, Germany, Lecture Notes in Computer Science, T.D.Wang et al. (Eds.): SEAL 2006, LNCS 4247, pp. 197-204, 2006.