

Stock Price Prediction Using Machine Learning Regression

Richard Jebraj, Student, EPCET, Bangalore India ,riche.jeb7@gmail.com

Mrs. Nandini Gowda P, Assistant Professor, EPCET, Bangalore, India ,nandini.epcet@gmail.com

Sai Sreeram B V, Student, EPCET, Bangalore, India ,saisriram027@gmail.com

Apoorva G, Student, EPCET, Bangalore, India ,apoorvaappu738@gmail.com

Rakshith H R, Student, EPCET, Bangalore, India ,rakshithgowdahr@gmail.com

Abstract: Time series forecasting is used to determine the future prices of stock, and the analysis and modeling of finance time series help investors in taking decisions and trades. In addition, in a dynamic environment such as the stock market, the non-linearity of the time series is pronounced, immediately affecting the efficacy of stock price forecasts. Hence, this work introduces an intelligent time series prediction system that uses sliding-window metaheuristic optimization for declaring the stock prices of companies in advance. It may be highly favourable to home brokers who do not have enough knowledge to invest in such companies. The system has a graphical user interface and functions as a stand-alone application. The developed compound system displayed outstanding prediction performance and it improves overall profit for investment performance. The proposed model is a promising predictive technique for highly non-linear time series, whose patterns are difficult to capture by traditional models.

Keywords—Construction company, data mining, prediction system, sliding-window, stock price forecasting, time series, machine learning, swarm intelligence and metaheuristic optimization.

I. INTRODUCTION

Financial markets change rapidly and generate huge amounts of data daily. Investment is a commitment of money or other resources to obtain benefits in the future. Stock is one type of securities. It is the most popular financial market instrument and its value changes quickly [1]. It can be defined as a sign of capital participation by a person or an enterprise in a company or a limited liability company.

The stock market provides opportunities for brokers and companies to make investments on neutral ground [2]. Stock prices are determined to regulate the future value of companies' stock or other financial instruments that are marketed on financial exchanges [3]. However, the stock market is characterized by nonlinearities, discontinuities, and high-frequency multi-polynomial components because it interacts with many factors such as political events, general economic conditions, and traders' expectations [4]. Therefore, making precise predictions of stock values are challenging.

Investors can buy stocks that are related to the construction firms that design infrastructure projects, hire contractors and handle paperwork, and decision-makers of construction firms can buy stocks from other companies. When the direction of the market is successfully predicted, investors may be better guided and monetary rewards will be substantial. The challenge in today's environment, where

bad news can always be heard, is to forecast proactively, rather than reactively. Therefore, construction corporations are trying to predict stock prices which is important to be considered on a financial exchange, against sudden drops in the market.

Time series forecasting consists in a research area designed to solve various problems, mainly in the financial area [5]. It is noteworthy that this area typically uses tools that assist in planning and making decisions to minimize investment risks. This objective is obvious when one wants to analyze financial markets and, for this reason, it is necessary to assure a good accuracy in forecasting tasks.

According to Saini (2016), forecasting based on a time series represents a means of providing information and knowledge to support a subsequent decision [6]. Thus, the analysis of time series focuses on achieving dependency relationships among historical data. The two broad categories of forecasting models are linear and nonlinear. For many decades, traditional statistical forecasting models in financial engineering were linear. Some well-known statistical models can be used in time series forecasting [6]. Conventional modeling techniques, such as the Box-Jenkins autoregressive integrated moving average, are not adequate for stock market price forecasting [7].

Machine learning (ML) is coming into its own that can play a key in a wide range of critical applications. In

machine learning, support vector machines (SVMs) have many advanced features that are reflected in their good generalization capacity and fast computation [6]. They are also not very sensitive to assumptions about error terms and they can tolerate noise and chaotic components. Notably, SVMs are increasingly used in materials science [8], the design of engineering systems [9] and financial risk prediction [10].

Support vector regression (SVR), a variant of the SVM, was developed by Vapnik et al. (1995) [11]. SVR is typically used to solve nonlinear regression problems by constructing the input-output mapping function. The least squares support vector regression (LSSVR) algorithm is a further development of SVR by Suykens (2001) [12] and involves equality instead of inequality constraints, and works with a least squares objective function. The LSSVR approach considerably reduces computational complexity and increases efficiency compared to standard SVR.

Recently, Lu et al. (2009) used independent component analysis to remove noise from forecasting variables. The filtered forecasting variables, which contain less noise information, then serve as the input variables of the SVR forecasting model [13]. Hao et al. (2006) examined the feasibility of methods in stock composite index forecasting and improved the accuracy of parameter selection by SVR. They concluded that SVR has high prediction performance [14].

Some studies have demonstrated the superiority of LSSVR over standard support vector regression (SVR) for estimating product cost [15] and energy utilization [16]. LSSVR solves linear equations instead of a quadratic programming problem. It is preferred for large-scale regression problems that demand fast computation [12]. Since time series data can be formulated by regression analysis, LSSVR is very efficient when applied to the issue at hand. However, the efficacy of LSSVR strongly depends on its tuning hyper parameters, which are the regularization parameter and the kernel function. Inappropriate settings of these parameters may lead to significantly poor performance of the model [17]. Therefore, the evaluation of such hyper parameters is a real-world optimization problem.

Since the performance of SVR-based models strongly depends on the setting of its hyperparameters, they used to be set in advance based on the experience of practitioners, by trial-and-error, or using a grid search algorithm [18]. Thus, finding the optimal values of regularization and kernel function parameters for SVR-based models is an important and time-consuming step. Therefore, a means of automatically finding the hyperparameters of SVR, while ensuring its generalization performance, is required.

Optimization is one of the cornerstones of science and engineering. Recently, the field of nature-inspired

optimization algorithms has grown incredibly fast. The algorithms are usually general-purpose and population-based. They are normally referred to as evolutionary algorithms because many of them are motivated by biological evolution. In a broad sense, evolutionary algorithms cover those that iteratively vary a group of solutions based on some nature-inspired operations.

Many evolutionary algorithms, such as artificial bee colony and cuckoo search (CS) algorithms, have been adopted to tune the hyper parameters of SVR [19, 20]. For instance, Jianzhou et al. (2014) designed compound smart forecasting models that were based on CS, the singular spectrum analysis, time series, and machine learning methods to conduct short-term power load prediction [20]. Hsieh et al. (2011) demonstrated that the particle swarm optimization (PSO)-based SVR model was superior to traditional SVR in forecasting the daily Taiwan Stock Exchange Capitalization Weighted Stock Index (TAIEX) [19].

The firefly algorithm (FA) [21], which is a nature-inspired metaheuristic method, has recently performed extremely well in solving various optimization problems such as stock price forecasting [18] and electricity price prediction [22]. The standard FA was developed by modeling the behavior of tropical fireflies. Notably, the smart firefly algorithm-based LSSVR has been demonstrated to be very effective in solving complex problems in civil engineering [23, 24].

FA and its variants are used to solve many optimization, classification and engineering problems. Gandomi et al. (2011) in [25] used FA to solve mixed continuous/discrete structural optimization problems in the design of welded beams, pressure vessels, helical compression springs, reinforced concrete beams, stepped cantilever beams, and car side impact design. The optimization results indicated that FA is more efficient than other meta-heuristic algorithms, such as PSO, genetic algorithm, simulated annealing and differential evolution. Recent research suggests that hybrid forecasting models can be usefully applied to the stock market's fluctuations, yielding satisfactory forecasting precision [4]. The authors used a hybrid model to capture the linear and non-linear characteristics of a stock price time series and confirmed that hybrid forecasting models are powerful tools for practitioners in management science. A review of the literature has indicated that enhancing the effectiveness capability of least squares support vector regression based on a nature-inspired metaheuristic optimization algorithm, such as the firefly algorithm [24, 26] is an unsolved problem in the field of stock price prediction.

Although artificial intelligence techniques and metaheuristic optimization algorithms are powerful, practitioners must be able to perform the extensive manual

operations. Users, such as traders in the financial market, are very interested in conveniently obtaining results that support their decisions. The computational cost of such advanced algorithms is high owing to their complexity. Therefore, creating an intelligent user system, which combines a knowledge base, a computing engine, and a graphical user interface, is a subject of great interest for many investors and financial analysts.

For example, Lee (2004) designed the iJADE Stock Advisor—an intelligent agent-based stock prediction system [27]. He integrated his proposed hybrid radial basis-function recurrent network (HRBFN) with the iJADE framework. Experimental results demonstrated that the HRBFN model can be successfully integrated with mobile-agent technology to provide a truly intelligent, mobile and interactive stock advisory solution.

This work develops an intelligent time series prediction system using sliding-window metaheuristic optimization. It involves the hybrid model of a metaheuristic firefly algorithm and least squares support vector regression (MetaFA-LSSVR) to forecast the prices of construction corporate stocks. The MetaFA is chosen as the training algorithm to enhance the efficiency of, and reduce the computational burden on the machine learner, LSSVR. The proposed expert framework is called the Intelligent Time Series Prediction System using Sliding-window Metaheuristic Optimization.

Section II presents the research methodology. Section III describes in detail application of the system. The final section provides concluding remarks and an outline for future work.

II. METHODOLOGY

A. Phase space reconstruction

In time series prediction, the time series are typically expanded into three or higher-dimensional space to exploit the information that is implicit in them. Selecting a suitable pairing of embedding dimension m (lag) and time delay τ is very important for phase space reconstruction [28]. The precision of m and τ is directly related to the accuracy of the constant of the described characteristics of the strange attractors in phase space reconstruction. The choices of delay time and embedding dimension are important, as good choices can reduce both the amount of data required and the effect of noise. In this study, the optimum values of the embedding parameters are determined by performing a sensitivity analysis.

Consider a time series $\vec{x} = \{x_1, x_2, \dots, x_N\}$. The time-delay vectors can be reconstructed as follows, where X is the input matrix and Y is the corresponding output matrix. The output of the analysis is fed back to the input and future values are predicted from previous values in the time series.

$$\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{bmatrix} \rightarrow X = \begin{bmatrix} x_1 & x_2 & \dots & x_{m-1} & x_m \\ x_2 & x_3 & \dots & x_m & x_{m+1} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ x_{N-m-\tau} & x_{N-m-\tau+2} & \dots & x_{N-\tau-1} & x_{N-\tau} \end{bmatrix}, Y = \begin{bmatrix} x_{m+\tau} \\ x_{m+\tau+1} \\ \vdots \\ x_N \end{bmatrix}$$

As suggest in [29], the learning dataset used in this study was collected within a sliding-window. Fig. 1 depicts the sliding-window and phase space construction. Since the forecast is one step ahead (hence the term, “one-step ahead forecasting”), the forecast horizon is 1. In the first validation, the working window includes p historical observations (x_1, x_2, \dots, x_p) , which are used to forecast the next value x_{p+1} .

In the second validation, the oldest value is x_1 removed from the window and the latest value x_{p+1} is added, keeping the length of the sliding window constant at p . The next forecast value will be x_{p+2} . The window continues to slide until the end of the dataset is reached. If the number of observations is N , then the total number of validations is $(N-p)$.

B. Metaheuristic optimization in machine regression learner

1) Regression model: Least squares support vector regression The LSSVR approach proposed by Suykens et al. (2002) [30] is a well-developed ML technique with many advanced features that support a high generalization capacity and fast computation. The LSSVR training process entails the use of a least squares cost function to obtain a linear set of equations in a dual space to minimize the computational cost. Accordingly, iterative methods, such as the conjugate gradient method are typically used to derive a solution by efficiently solving a set of linear equations. To reduce the computational burden of the LSSVR for function estimation, the regression model in this study uses a quadratic loss function [29].

In a function estimation of the LSSVR, the optimization problem is formulated as

$$\min_{w,b,e}(\omega, e) = \frac{1}{2} \|\omega\|^2 + \frac{1}{2} C \sum_{k=1}^N e_k^2$$

(2) Since this is a typical optimization problem of a differentiable function with constraints, it can be solved by using Lagrange multipliers. Equation (3) is the resulting LSSVR model for function prediction.

$$f(x) = \sum_{k=1}^N a_k K(x, x_k) + b \tag{3}$$

where a_k and b are Lagrange multipliers and the bias term, respectively; and $K(x, x_k)$ is the kernel function. In this study, a radial basis function kernel (RBF) is used. For a detailed technical explanation of this approach applied in this study, the interested readers can refer to the work of Chou et al. (2016) [24]. However, the prediction accuracy

of the LSSVR is highly dependent on the determination of its hyperparameters. Therefore, as a part of this study, the enhanced FA algorithm was developed to optimized LSSVR hyperparameters, i.e., the regularization parameter (C) and the sigma of the RBF kernel (σ). 2) Tuning hyperparameters: swarm and metaheuristic optimization algorithm The FA, developed by Yang [21], is among the most successful swarm intelligence methods. This algorithm was inspired by the flashing patterns and behavior of tropical fireflies. For a maximization problem, the brightness is simply set to be proportional to the value of the objective function. Since the attractiveness of a firefly is proportional to the intensity of its light that is visible to adjacent fireflies, the attractiveness β of a firefly satisfies (4)

$$\beta = \beta_0 e^{-\gamma r^2}$$

where β is the attractiveness of the firefly; β_0 is the attractiveness of the firefly at $r=0$; r is the distance between the firefly of interest and any other; e is a constant coefficient; and γ is the absorption coefficient. The detailed FA procedure has been summarized in [8]. Although the FA is highly efficient in many applications, it often becomes trapped in a local optimum [31]. Moreover, setting tuning parameters that improve the convergence of the FA is another challenge. The FA control parameters should be optimized to balance exploitation and exploration. Therefore, the MetaFA incorporates three metaheuristic components, namely chaotic map, the adaptive inertia weight, and Lévy flight, in the conventional FA to enhance its search and optimization capabilities [32]. Fig. 2 describes the pseudocodes for the MetaFA-LSSVR model.

Logistic chaotic map for enhancing initial population

The FA uses a typical approach to generating an initial solution at random. The two major disadvantages of this approach are its slow convergence and its tendency to become trapped in local optima because of reduced population diversity. To improve initial diversity of solutions and the quality of the initial population, a logistic chaotic map is used to generate a highly diverse population of fireflies in the initial stage.

Gauss/mouse chaotic map for tuning attractiveness

The Gauss/mouse map provides the best means of tuning the attractiveness parameter (β) of the FA. For a detailed technical explanation of this approach applied in this study, the interested readers can refer to the work of He et al. (2001) [33].

Adaptive inertia weight for adjusting randomization

Reducing randomness as the iterations proceed improves the convergent efficiency of a swarm-based algorithm. In the initial stages of the search process, a large inertia weight

can boost global exploration performance (searching of a new area). In each of the final stages, reducing the inertia weight enhances local exploration (fine tuning of the current search area). Inertia weight is an essential in the convergence of the optimal known solutions to the globally optimal value; inertia weight also improves the execution time of the simulation.

Lévy flight for controlling movement

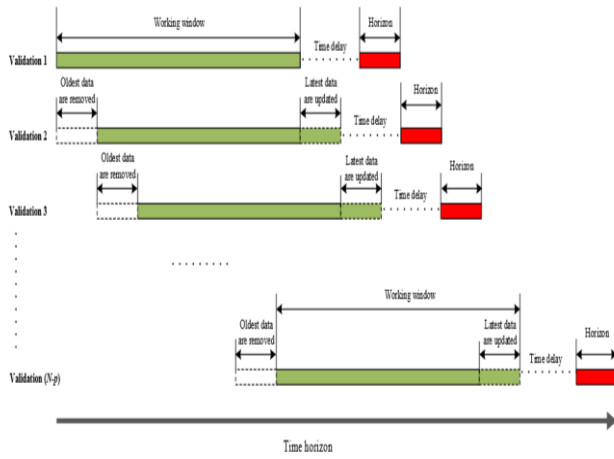
Random walk theory plays a critical role in modern swarm intelligence and evolutionary optimization algorithms [34]. Lévy flights are a random walk in which the step length is a Lévy distribution. The step lengths have no characteristic scale as the second moment or even the first moment may diverge, and the distribution exhibits self-affine properties. Lévy flights are used to generate random numbers in two steps: random selection of a direction and generation of steps that obey the selected Lévy distribution. In this work, directions were generated with uniform distributions. The Mantegna algorithm, which is the most popular, is used to generate steps from a symmetric Lévy stable distribution.

Intelligent time series prediction system using sliding-window metaheuristic optimization

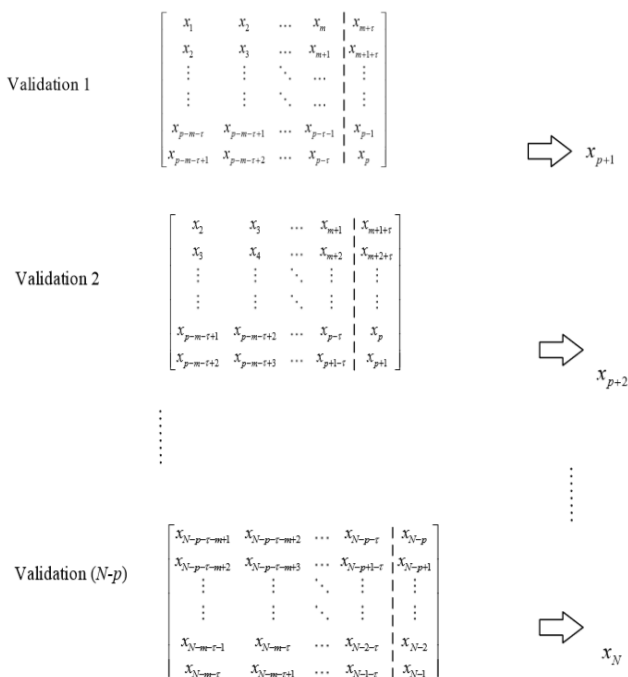
The development tools that are used in this study are MATLAB GUIDE, MATLAB compiler and MATLAB compiler runtime, all of which are based on MATLAB software and developed in a Window environment on a machine with an Intel Core i5 and 4GB of RAM. MATLAB GUIDE was used to design a user-friendly interface. The proposed forecasting model has two main stages. In the first stage, the values of the time series parameters, lag and the size of the sliding-window, are determined. Learning data in the sliding-window are incorporated into input and output matrices. In the second stage, the MetaFA-LSSVR model is used to perform one-step ahead forecasting. Each validation involves one-step ahead forecasting, so the number of test data equals the number of validations. The window continues to move ahead and validations are conducted. The process is repeated until all validations have been performed.

There are two modules, Evaluation and Forecast, provided in the interface system.

The Evaluation module examines the performance of the sliding-window MetaFA-LSSVR and LSSVR models. For evaluation, the user can choose from several options, which are use the opened data file, use the test file, hold-out, and sliding-window validation. The Forecast module performs one- or multi- day ahead forecasting. The system returns a performance evaluation or predicted values, as required. The system also provides a feature that allows the user to save the model after it is run, enabling the user to reuse it for another purpose.



(a) sliding window representation



(b) Phase space reconstruction for time series analysis
Fig. 1. Sliding Window and phase space reconstruction

Fig. 3 displays the architecture of the proposed intelligent time series prediction system with sliding-window metaheuristic optimization. The user of the system involves the following simple steps - parameter setting, normalization, optimization, and purpose with MetaFA-LSSVR component. The results are shown in the interface. The user can also read the analysis report or save results to an electronic file for further analysis.

Equation (5) is the fitness function of the MetaFA-LSSVR

$$f(m) = \text{objective_function}_{\text{validation-data}}$$

in which the objective function can be designated as the root mean square error, the mean absolute error, the mean absolute percentage error, the non-linear regression multiple correlation coefficient or the mean square error. However, if the data set contains actual zero values, then the mean

absolute percentage error cannot be used as an objective function.

```

Perform objective function  $f(x)$ ,  $x = (x_1, \dots, x_d)^T$ 
Set search space and number of generations
Generate initial population of fireflies  $x_i$ , ( $i=1, 2, \dots, n$ ) using logistic chaotic map
Determine light intensity  $I_i$  at  $x_i$  by  $f(x_i)$ 
Define light absorption coefficient  $\gamma$ 
Generate initial population,  $k = 0$ 
1. While ( $t \leq \text{MaxGeneration}$ ) do
    (1) Update the generation number,  $k = k + 1$ 
    (2) Tune randomization parameter  $\alpha$  by adaptive inertia weight ( $\alpha = \alpha_0, 0.9^k$ )
    (3) Tune attractiveness parameter  $\beta$  by using Gauss/mouse chaotic map
    for  $i = 1$ : No. fireflies
        for  $j = 1$ : No. fireflies
            if ( $I_j > I_i$ )
                Move firefly  $i$  toward  $j$  in  $d$ -dimension by Lévy flight
            end if
            Vary attractiveness with distance  $r$  via  $\exp[-\gamma * r]$ 
            Evaluate new solutions and update light intensity
        end for  $j$ 
    end for  $i$ 
    Rank the fireflies and find the current best
end while
2. MetaFA-LSSVR function validation
   Set kernel (rbf) and loss-function (least-square) parameters
   Train model with hyperparameters ( $C, \sigma$ )
   Evaluate trained LSSVR model
   Evaluate fitness function  $f(m)$ , and go to step 1
3. Has the stopping criterion been met?
   If the criterion has been met,
       Go to step 4
   else
       go to step 1
End
4. Optimized LSSVR model
5. Postprocess results and visualization

```

Fig. 2. Pseudo Code for MetaFA-LSSVR model

III. SYSTEM APPLICATIONS

A. Data collection

Historical daily prices were taken from Yahoo! Finance, a publicly accessible website, as they were by Xiong et al. (2014) [18]. Six years (5th October, 2011 to 31st May 2017) of daily data on five stocks - Yuanta/P-shares Taiwan Top 50 ETF (0050.TW) Highwealth Construction Corp. (2542.TW), Huang Hsiang Construction Corporation (2545.TW), Ruentex Engineering & Construction Co., Ltd. (2597.TW) and Chong Hong Construction Co., Ltd. (5534.TW) – were downloaded from Yahoo! Finance. The data were closing stock prices. The 0050.TW stock dataset includes top 50 stocks by market capitalization in Taiwan. 2542.TW, 2545.TW, 2597.TW, and 5534.TW stocks provide important indicators of the overall performance of the construction market in Taiwan. Table II presents details about the stocks and the number of data instances for each stock.

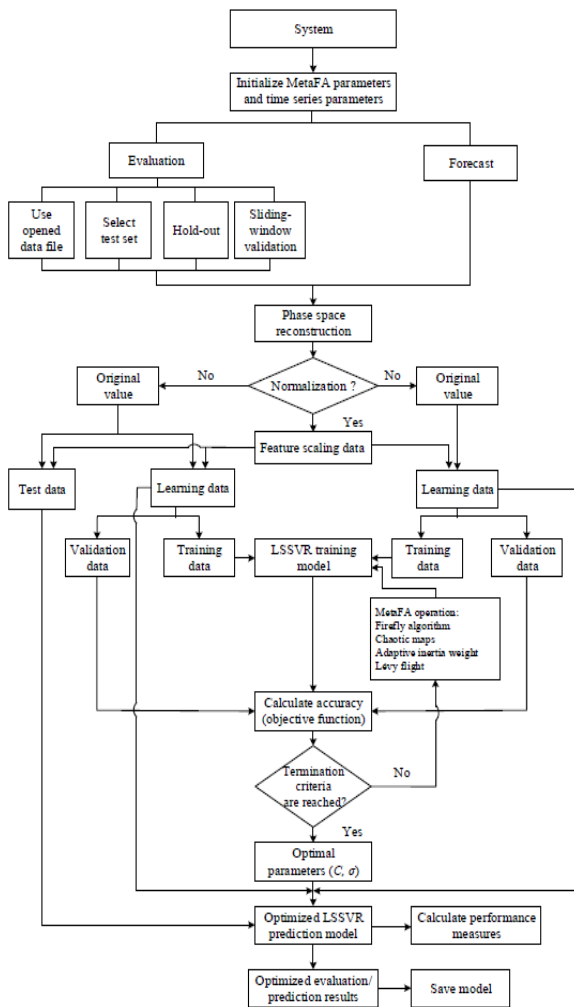


Fig. 3. System flowchart

B. Input setup

The performance of the proposed model in short-term prediction over the next 1, 10, and 15 days and in long-term prediction over the next 30, 60, 90, 180, and 360 days is evaluated using five stock datasets. Its performance is compared with those of previously reported models. Then, its performance is tested using other popular stock datasets that are available in literature. Even though this investigation focuses mostly on short-term price prediction, long-term price predictions were carried out. Part of the default system settings were as follows in [8]. The number of learning/training data was set equal to the size of the sliding-window. The size of the sliding window was set to 840. Based on several trials and experiments, the optimal lag was 2.

C. Analysis of experimental results

1) Results of proposed hybrid prediction system

Eight scenarios were compared (i.e., 1, 10, 15, 30, 60, 90, 180 and 360 days ahead of time). Table III presents results for the 0050.TW stock. Fig. 4 displays the predicted closing prices of the 0050.TW stock dataset for the specified days ahead. This figure indicates that values that were predicted one day ahead by the system were closer to the actual values

than the others. Table IV to VII present similar results for 2542.TW, 2545.TW, 2597.TW and 5534.TW stocks.

TABLE I
MATHEMATICAL FORMULAS FOR PERFORMANCE MEASURES

Measure	Formula
Root mean square error (RMSE)	$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y'_i - y_i)^2}$
Mean absolute error (MAE)	$MAE = \frac{1}{n} \sum_{i=1}^n y'_i - y_i $
Mean absolute percentage error (MAPE)	$MAPE = \frac{1}{n} \sum_{i=1}^n \left \frac{y'_i - y_i}{y_i} \right $
Mean square error (MSE)	$MSE = \frac{1}{n} \sum_{i=1}^n (y'_i - y_i)^2$
The correlation coefficient (R)	$R = \frac{n \sum y_i y'_i - (\sum y_i)(\sum y'_i)}{\sqrt{n(\sum y_i^2) - (\sum y_i)^2} \sqrt{n(\sum y'^2) - (\sum y')^2}}$
Non-linear regression multiple correlation coefficient (R^2)	$R^2 = 1 - \frac{\sum_{i=1}^n (y_{i+1} - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2}$
Synthesis index (SI)	$SI = \frac{1}{m} \sum_{i=1}^m \left(\frac{P_i - P_{\min,i}}{P_{\max,i} - P_{\min,i}} \right)$

TABLE II
DESCRIPTION OF ANALYZED STOCKS AND DATASETS

Dataset	Total No. of data points	Duration
0050.TW	1357	Oct 5, 2011 to May 31, 2017
2542.TW	1457	Oct 5, 2011 to May 31, 2017
2545.TW	1457	Oct 5, 2011 to May 31, 2017
2597.TW	1457	Oct 5, 2011 to May 31, 2017
5534.TW	1457	Oct 5, 2011 to May 31, 2017

The above results clearly reveal that the error increases with the number of days in advance for which the prediction is made. This may be obvious for any prediction system. Over all, the proposed system yielded better predictions for the 0050.TW stock dataset than the other stock datasets. For instance, one-day ahead prediction using the 0050.TW stock dataset yielded favorable overall performance measures. However, the RMSE, MAE, MAPE, and MSE values of the 360 day-ahead prediction were smaller than those of the 90 day-ahead prediction and the 180 day-ahead prediction (with an RMSE of 4.763, an MAE of 4.086, an MAPE of 5.936 % and an MSE of 22.690). This will motivate future research for long-term investment.

For the 2597.TW stock dataset, the one day-ahead forecast of the stock price was the best with an RMSE of 1.548, an MAE of 0.617, an MAPE of 1.372%, an R of 0.990, an R2 of 0.973 and an MSE of 2.396. Table VI also reveals that RMSE, MAE, MAPE and MSE values of the 360 day-ahead prediction were smaller than those of the 180 day-ahead prediction (with an RMSE of 16.333, an MAE of 8.608, an MAPE of 19.948 % and an MSE of 266.769). Similar results were obtained using the other datasets.

2) Comparison of profit

Statistical performance measurements and direction of stock price do not have much meaning for practical investors. The financial performance of a forecasting model must also be examined to evaluate a forecasting model. The buying and selling behaviors of a typical investor can be simulated. An investor will buy stocks from the market if he/she expects an increase in prices and sell his/her financial assets to prevent a potential loss if he/she expects a decrease in prices. This simple trading logic was simulated using the predictions made using the proposed model. For simplicity, no tax or fees were associated with any transaction.

According to previous research, the profit that was earned on the eight stocks during 300 trading days when investments were managed according to the forecasts made using the proposed system. Table XI compares the results of trading using the proposed hybrid system with those obtained using the BUY and HOLD method. In this method, an investor buys stocks from the beginning of trading period closing price and sells all of its assets from the end of trading period closing price. The proposed system outperforms with the BIST 100 stock. More profit was made on the 2542.TW stock than on the other construction company stocks. Table XI also reveals that the proposed system with the eight stocks yields greater profits than the traditional BUY and HOLD method.

IV. CONCLUSION

Decision to buy or sell a stock is very complicated since many factors can affect stock price. This work presents a novel approach, based on a metaheuristic firefly algorithm and least squares support vector regression (MetaFA-LSSVR), to constructing a stock price forecasting expert system, with the aim of improving forecasting accuracy. The intelligent time series prediction system that uses sliding-window metaheuristic optimization is a graphical user interface that can be run as a stand-alone application. The system makes the prediction of stock market values simpler, involving fewer computations, than that using the other method that was mentioned above. The original FA is supplemented with three metaheuristic components – chaotic maps, adaptive inertia weight, and Lévy flight - to construct a metaheuristic optimization algorithm (MetaFA). The superior performance of the MetaFA was verified by

validating benchmark functions. Thus, the MetaFA was adopted to tune automatically the hyperparameters C and σ of the LSSVR. The optimized LSSVR prediction model was used with the sliding-window approach to evaluate and forecast stock price. Default settings of the system, including pre-defined parameters, save users time. To evaluate the proposed approach, it was applied to five datasets for stocks in Taiwan, and three other stock datasets that have been used in other papers. Statistical measures were obtained when applied to the performance of the proposed approach. Taiwan construction company stock datasets achieved in forecast 1, 15 and 30 days ahead of time at the desired statistical significance levels. In particular, the one day prediction of 2597.TW stock prices was better than that of any construction company stock prices with an MAPE of 1.372%, an R of 0.990, and an R2 of 0.973. Toward the end of the study, the financial performance of the proposed system was examined with encouraging results. Therefore, the proposed system can be used as a decisive tool to forecast stock prices for short-term investing. This study focuses on the stock price market in Taiwan. To generalize the application of the proposed system, future work should use the proposed system to estimate other stocks in similar emerging markets and mature markets, such as Vietnam, Indonesia, China, Japan, Hong Kong, Korea, Singapore, Europe, and USA. Secondly, the system can be extended to analyze multivariate time series data and import raw dataset directly. Thirdly, profit can be maximized even when the construction corporate stock market is bullish. Finally, the development of a web-based application should be considered to improve the user-friendliness and usability of the expert system. The limitation of the proposed system is its computational speed, especially with respect to sliding-window validation, because of the complexity of solving large mathematical loops in the MATLAB program. The computational cost increases with the number of validations. Another weakness is the need to define many parameters of the system (MetaFA and time series parameters) though the default settings are provided. Moreover, the system does not achieve outstanding results for long-term investment – a finding that will motivate future research.

REFERENCES

- [1] J. P. He, L. Cai, P. Cheng, and J. L. Fan, "Optimal Investment for Retail Company in Electricity Market," *Ieee Transactions on Industrial Informatics*, vol. 11, no. 5, pp. 1210-1219, Oct, 2015.
- [2] Y. Perwej, A. Perwej, and "Prediction of the Bombay Stock Exchange (BSE) market returns using artificial neural network and genetic algorithm," *Journal of Intelligent Learning Systems and Applications*, vol. 4, pp. 108-119, 2012.

- [3] A. A. Arifah, and I. Mukhlash, "The Application of Fuzzy Association Rule on Co-movement Analyze of Indonesian Stock Price," *Procedia Computer Science*, vol. 59, pp. 235-243, 2015/01/01, 2015.
- [4] M. Göçken, M. Özçalıcı, A. Boru, and A. T. Dosdoğru, "Integrating metaheuristics and Artificial Neural Networks for improved stock price prediction," *Expert Systems with Applications*, vol. 44, pp. 320-331, 2016.
- [5] Y. Bao, Y. Lu, and J. Zhang, "Forecasting Stock Price by SVMs Regression," *Artificial Intelligence: Methodology, Systems, and Applications*, C. Bussler and D. Fensel, eds., pp. 295-303, Berlin, Heidelberg: Springer Berlin Heidelberg, 2004.
- [6] D. Saini, A. Saxena, and R. C. Bansal, "Electricity price forecasting by linear regression and SVM." pp. 1-7.
- [7] J. Wang, R. Hou, C. Wang, and L. Shen, "Improved v - Support vector regression model based on variable selection and brain storm optimization for stock price forecasting," *Applied Soft Computing*, vol. 49, pp. 164-178, 2016.
- [8] J.-S. Chou, and N.-T. Ngo, "Engineering strength of fiber-reinforced soil estimated by swarm intelligence optimized regression system," *Neural Computing and Applications*, December 28, 2016
- [9] A. Jindal, A. Dua, K. Kaur, M. Singh, N. Kumar, and S. Mishra, "Decision Tree and SVM-Based Data Analytics for Theft Detection in Smart Grid," *Ieee Transactions on Industrial Informatics*, vol. 12, no. 3, pp. 1005-1016, Jun, 2016.
- [10] V. Behbood, J. Lu, and G. Q. Zhang, "Fuzzy Refinement Domain Adaptation for Long Term Prediction in Banking Ecosystem," *Ieee Transactions on Industrial Informatics*, vol. 10, no. 2, pp. 1637-1646, May, 2014.
- [11] V. N. Vapnik, *The nature of statistical learning theory*: Springer-Verlag New York, Inc., 1995.
- [12] J. A. K. Suykens, "Nonlinear modelling and support vector machines." pp. 287-294.
- [13] C.-J. Lu, T.-S. Lee, and C.-C. Chiu, "Financial time series forecasting using independent component analysis and support vector regression," *Decision Support Systems*, vol. 47, no. 2, pp. 115-125, 2009.
- [14] W. Hao, and S. Yu, "Support Vector Regression for Financial Time Series Forecasting," *Knowledge Enterprise: Intelligent Strategies in Product Design, Manufacturing, and Management*, K. Wang, G. L. Kovacs, M. Wozny and M. Fang, eds., pp. 825-830, Boston, MA: Springer US, 2006.
- [15] S.-G. Deng, and T.-H. Yeh, "Using least squares support vector machines to the product cost estimation," *Journal of Chung Cheng Institute of Technology*, vol. 40, no. 1, pp. 1-16, 2011.
- [16] F. Kaytez, M. C. Taplamacioglu, E. Cam, and F. Hardalac, "Forecasting electricity consumption: A comparison of regression analysis, neural networks and least squares support vector machines," *International Journal of Electrical Power and Energy Systems*, vol. 67, pp. 431-438, 2015.
- [17] K. Duan, S. S. Keerthi, and A. N. Poo, "Evaluation of simple performance measures for tuning SVM hyperparameters," *Neurocomputing*, vol. 51, pp. 41-59, 2003.
- [18] T. Xiong, Y. Bao, and Z. Hu, "Multiple-output support vector regression with a firefly algorithm for interval-valued stock price index forecasting," *Knowledge-Based Systems*, vol. 55, pp. 87-100, 2014.
- [19] H.-I. Hsieh, T.-P. Lee, and T.-S. Lee, "A Hybrid Particle Swarm Optimization and Support Vector Regression Model for Financial Time Series Forecasting," *International Journal of Business Administration*, vol. 2, pp. 48-56, 2011.
- [20] W. Jianzhou, J. Shiqiang, Q. Shanshan, and J. Haiyan, "Swarm Intelligence-Based Hybrid Models for Short-Term Power Load Prediction," *Mathematical Problems in Engineering*, vol. 2014, pp. 17, 2014.
- [21] X.-S. Yang, *Nature-Inspired Metaheuristic Algorithms*: Luniver Press, 2008.
- [22] A. K. Rout, R. Bisoi, and P. K. Dash, "A low complexity evolutionary computationally efficient recurrent Functional link Neural Network for time series forecasting," 2015 IEEE Power, Communication and Information Technology Conference, PCITC 2015 - Proceedings. pp. 576-582.