Çukurova Üniversitesi Mühendislik Mimarlık Fakültesi Dergisi, 31(ÖS 2), ss. ÖS 53-ÖS 61, Ekim 2016
Çukurova University Journal of the Faculty of Engineering and Architecture, 31(SI 2), pp. SI 53-SI 61, October 2016

A Hybrid Extreme Learning Machine and its Variant for Stock Price Prediction

Mustafa GÖÇKEN[*1], Mehmet ÖZÇALICI[2], Aslı BORU[1], Ayşe Tuğba DOSDOĞRU[3]

[1]*Adana Science and Technology University, Industrial Engineering Department, Adana*
[2]*Kilis 7 Aralık University, Business and Administration Department, Kilis*
[3]*Gaziantep University, Industrial Engineering Department, Gaziantep*

Geliş tarihi: 10.07.2016   Kabul tarihi: 24.09.2016

Abstract

Accurate and effective stock price prediction is appealing for investors due to the potential of obtaining a very high return. However, it is still a challenging task in the modern business world because of the complex, evolutionary, and nonlinear nature of stock market. Therefore, we proposed two hybrid models, which are Harmony Search (HS) based Extreme Learning Machine (ELM) that is denoted as HS-ELM and HS based Recurrent Extreme Learning Machine (RELM) that is represented as HS-RELM, to provide accurate and fast one-day ahead stock price prediction. This study provides a new direction in the field of stock price prediction and offers some suggestions on how to configure HS-ELM and HS-RELM for performing stock price prediction, with an application on stocks listed in BIST50 Index. The results of the performance measures show that although both proposed models are very helpful for the practical applicability of the stock market, HS-RELM model is more powerful than HS-ELM model.

Keywords: Extreme learning machine, Recurrent extreme learning machine, Harmony search, Stock price prediction

Melez Aşırı Öğrenme Makinesi ve Türevi ile Hisse Senedi Fiyatı Tahmini

Öz

Çok yüksek getiri elde etme potansiyeline sahip olması nedeniyle doğru ve etkili hisse senedi fiyatı tahmini yatırımcılar için caziptir. Bununla birlikte, borsanın karmaşık, evrimsel ve doğrusal olmayan yapısı nedeniyle, modern iş dünyasında hâlâ karmaşık bir iştir. Bu nedenle, iki melez model, HS-ELM olarak adlandırılan Harmoni Araması (HS) tabanlı aşırı öğrenme makinesi (ELM) ve HS-RELM olarak adlandırılan HS tabanlı tekrarlı aşırı öğrenme makinesi (RELM), günlük hisse senedi fiyatı tahminini doğru ve hızlı bir şekilde elde etmek için önerilmiştir. Bu çalışma, hisse senedi fiyatı tahmini alanına yeni bir yön vermekte ve BIST50 Endeksinde bulunan farklı hisse senetleri üzerinde uygulanması ile HS-ELM ve HS-RELM'nin hisse senedi fiyat tahmininde nasıl yapılandırılması gerektiği konusunda bazı öneriler

[*] Sorumlu yazar (Corresponding author): Mustafa GÖÇKEN, *mgocken@adanabtu.edu.tr*

sunmaktadır. Performans ölçümlerinin sonuçları, her iki önerilen modelin hisse senetleri fiyat tahminine pratik uygulanabilirliği açısından oldukça yararlı olduğunu göstermesine rağmen HS-RELM modelinin performansının HS-ELM modelinin performansından daha iyi olduğu gözlemlenmiştir.

Anahtar Kelimeler: Aşırı öğrenme makinesi, Tekrarlı aşırı öğrenme makinesi, Harmoni araması, Hisse senedi fiyatı tahmini

1. INTRODUCTION

Over the years, stock markets are an important part of the nations' economy since the greatest amount of capital is exchanged implicitly through stock markets all over the world [1]. In this case, analyzing stock market performance has become reasonably challenging due to the effect of economic globalization and information technology. One of the most common characteristics for all stock markets is the uncertainty related to their future states. This is undesirable for the investors, but unavoidable when stock market is chosen for investment. At this point, decisions on investment are important and entail prediction [1]. The prediction largely affects the profitability of investing and trading in the stock market [2]. Therefore, academicians and practitioners have primarily investigated the various prediction models for years. In the light of previous studies, it can be said that although many models have been employed for predicting stock price, each model has its own advantages and disadvantages under different situations. Also, we determined three major drawbacks. Firstly, various input variables are used to predict stock price based on personal experience that can lead to wrong judgement on investment decisions. Secondly, multi-feature data typically generate high-dimensional data and cause higher computational complexity. Thirdly, specific assumptions are needed for statistical models that cannot be implemented to the datasets which do not satisfy these assumptions [3]. At this point, extreme learning machine (ELM) can be considered as one of the most powerful models to overcome difficulties encountered by other models. For example, ELM avoids many of the difficulties faced by gradient-based learning models such as learning rate and epochs, stopping criteria, and local minima while producing higher generalization performance [4]. Details can be found in Huang et al. [5]. Zhu et al. [4] presented a hybrid learning algorithm where the output weights of ELM are analytically assigned by using Moore–Penrose (MP) generalized inverse while the input weights and hidden biases of ELM are randomly determined by modified differential evolution. The result of study shows that proposed model can provide a much more compact network. Bazi et al. [6] applied the differential evolution to analyze the model selection problem of ELM. Yang et al. [7] proposed evolutionary based ELM with differential evolution to balance the explorative power and exploitive power and to reduce the prediction time of original ELM. Suresh et al. [8] presented a real-coded genetic algorithm to determine the input weights, optimal hidden neuron numbers, and bias values of ELM. In the study, proposed model gets a compact network with better generalization but it causes a higher computational effort. To solve this problem, a sparse-ELM is used to choose the number of neurons, input weights and bias values quickly in ELM. Hegazy et al. [9] used flower pollination algorithm and ELM for monthly stock price prediction. In the study, flower pollination algorithm is utilized to select input weights and hidden biases to create more compact network structure than traditional ELM model. Application of ELM on stock market prediction can also be found in Wang et al. [10] and Li et al. [11].

In recent years, recurrent ELM (RELM) is also used to improve the prediction performance. The major difference between RELM and ELM is that RELM is built for sequential (time ordered) datasets but the order of the data in the dataset is not important in ELM. In addition, the number of neurons in the input layer of ELM is the same with the number of features in the dataset while the number of neurons in the input layer of RELM is the sum of the number of features and the number of context neurons [12]. Application of RELM on electricity load forecasting can be found in [12]. Considering previous research, this study is basically aimed to show how the use of HS together with ELM and RELM models allows one to obtain considerable prediction accuracy and to compare the performance of the proposed models in three stocks listed in BIST50 for emphasizing the importance of the specific parameter setting in HS-ELM and HS-RELM.

To the best of our knowledge, this study applies RELM on predicting one-day ahead stock price in BIST50 for the first time. In addition, we strengthen RELM and ELM models by integrating them with HS (Figure 1).

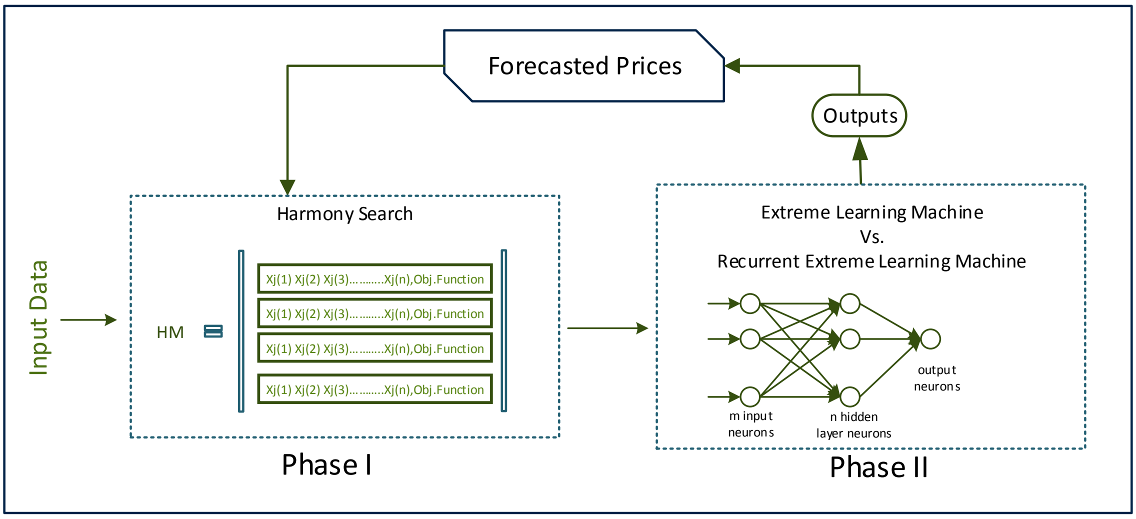

Figure 1. Hybrid ELM approach

Thus, hybrid stock price prediction models including HS-ELM and HS-RELM are proposed to provide a new perspective on developing ELM and RELM models. The advantage of these hybrid models lies in synergy between the HS and ELM (RELM) model. HS is used in Phase I to get the optimal parameters of the HS-ELM and HS-RELM models including input variables (technical indicators), activation function type, number of neurons in hidden layer and also number of neurons in the context layer for RELM. The results of HS are evaluated according to performance measure. In Phase II, the best results of HS are fed in proposed hybrid models. This procedure is repeated until desired iteration is reached in ELM and RELM. If desired iteration is not satisfied, HS is reused to obtain the better result.

The rest of the paper is organized as follows. Section 2 describes the basic structure of HS, ELM, and RELM models. In Section 3, analysis of proposed HS-ELM and HS-RELM models is given. Finally, conclusions are expressed in Section 4.

2. PROPOSED MODELS

2.1. Harmony Search

Metaheuristics are known as one of the most practical models to solve many complex optimization problems. The practical advantages of these models are their effectiveness and general applicability because many optimization models have failed to be either efficient or effective. Therefore, metaheuristics are generally preferred over other optimization models to find the solutions with many local optima and little inherent structure to guide the search [13]. In this study, we used Harmony Search (HS) to optimize the structure of ELM and RELM. HS is inspired by the music improvisation process. The major advantages of HS are that it does not need to specify the initial value settings and complex mathematical calculations. Basic steps of HS can be given as follows [14]:

Step 1. Harmony Memory Initialization: A harmony memory matrix (HMM) is generated and initialized. It includes a specified number of solutions that are also known as harmony memory size (*hms*). Each solution (harmony vector, I^i) consists of m integer numbers between 1 to N that are selected randomly and each of which corresponds to the sequence number of design variables in the design pool, and is displayed in a separate row of the matrix; consequently the size of HMM is $hms \times m$ (Eq. (1)).

$$\mathrm{HMM}=\begin{pmatrix} I_1^1 & I_2^1 & I_3^1 & \cdots & I_m^1 & Ø(I^1) \\ \vdots & & \ddots & & & \vdots \\ I_1^{hms} & I_2^{hms} & I_3^{hms} & \cdots & I_m^{hms} & Ø(I^{hms}) \end{pmatrix} \quad (1)$$

I_i^j is the sequence number of the ith design variable in the jth randomly selected feasible solution.

Step 2. Harmony Memory Evaluation: In this step, HS model analyzed hms solutions whose objective function values are computed. According to each objective function value in ascending order of magnitude, the solutions are sorted in the HMM.

Step 3. A New Harmony Improvisation: HS utilizes a stochastic random search instead of a gradient search. Note that stochastic random search is based on harmony memory considering rate ($hmcr$) and the pitch adjusting rate (par) so that derivative information is unnecessary [15]. $hmcr$, which is the probability of selecting a component from the harmony memory, is used in improvisation process. For each variable, a random number r_i is produced between 0 and 1 to execute this probability. According to r_i, either harmony memory (if $r_i \leq hmcr$) or the entire discrete set (if $r_i > hmcr$) is used to select each design variable for a new harmony ($I' = I_1',\ I_2', \ldots, I_m'$) improvisation as given in Eq. (2).

$$I_i' = \begin{cases} I_i' \in \{I_i^1, I_i^2, \ldots, I_i^{\mu}\} & \text{if } r_i \leq hmcr \\ I_i' \in \{1, \ldots, N\} & \text{if } r_i > hmcr \end{cases} \quad (2)$$

After the $hmcr$, each decision variable is evaluated to decide whether pitch-adjusted or not. At this point, par parameter (Eq. 3) is used to carry out this evaluation. par can be defined as sampling the variable's one of the neighboring values, obtained by adding or subtracting one from its current value. Note that the value of the variable does not change when it is not activated by par (Eq. (3)).

$$I_i'' = \begin{cases} I_i' \mp 1 & \text{if } r_i \leq par \\ I_i' & \text{if } r_i > par \end{cases} \quad (3)$$

Step 4. Update of Harmony Matrix: The objective function value is used to evaluate the newly generated harmony vector. If this value is better than the objective function value of the worst harmony vector in HMM, the existing worst harmony vector is excluded from the HMM and the newly generated harmony vector is included in the HMM.

Step 5. Termination: Steps 3 to Step 4 are repeated until desired iteration is reached. Finally, the best solution is chosen from the final HMM. HS parameter values for proposed models are given in the Table 1.

Table 1. HS parameter values

hms	50
hmcr	0.95
par	0.3
Maximum number of iteration	50

Determining the most relevant input variables (technical indicators) is a primary task of stock price prediction models. In addition, some essential modifications are required to provide more accurate prediction. Therefore, we used HS for the selection of the most relevant technical indicators and activation function type in ELM and RELM. Furthermore, HS is used for the determination of number of neurons in hidden layer in ELM and RELM. Apart from these, the number of neurons in the context layer is determined by HS. In literature, trial and error models are generally used to determine these parameters but this model is very time consuming and error-prone. Therefore, proposed HS-ELM and HS-RELM provide a new perspective on stock price prediction. The initial feature pool is given in Table 2.

Table 2. Initial feature pool

Index	Feature Name	Index	Feature Name
1	Previous Close	23	Close Price Accelerator
2	Previous High	24	Opening Price Momentum
3	Previous Low	25	High Price Momentum
4	Previous Open	26	Low Price Momentum
5	5 Day Simple Moving Average	27	Close Price Momentum
6	6 Day Simple Moving Average	28	Chaikin Volatility
7	10 Day Simple Moving Average	29	K% Stochastic
8	20 Day Simple Moving Average	30	D% Stochastic
9	5 Day Exponential Moving Average	31	Slow K% Stochastic
10	6 Day Exponential Moving Average	32	Slow D% Stochastic
11	10 Day Exponential Moving Average	33	William %R
12	20 Day Exponential Moving Average	34	Relative strength Index
13	5 Day Triangular Moving Average	35	Middle Bollinger Band
14	6 Day Triangular Moving Average	36	Upper Bollinger Band
15	10 Day Triangular Moving Average	37	Lower Bollinger Band
16	20 Day Triangular Moving Average	38	Highest High Value
17	Accumulation Distribution Oscillator	39	Lowest Low Value
18	Closing Price MACD	40	Median Price
19	9 Day Moving average of Close MACD	41	Price Rate of Change
20	Opening Price Accelerator	42	Typical Price
21	High Price Accelerator	43	Weighted Close
22	Low Price Accelerator	44	William’s Accumulation/Distribution Line

Table 3. The parameter values in proposed hybrid models

	Model type	Activation Function Type	Number of neurons in hidden layer	Number of neurons in the context layer	Index of Selected features				
AKSEN	HS-RELM	1	16	17	15	12	32	18	8
	HS-ELM	4	8	-	11	25	6	8	36
DOHOL	HS-RELM	1	21	16	11	33	4	40	37
	HS-ELM	4	16	-	15	7	24	21	2
SISE	HS-RELM	1	21	16	11	33	4	40	37
	HS-ELM	4	16	-	15	7	24	21	2

In HS, the number of input variables is fixed at 5 to receive fast result from HS. Thus, five features are selected from initial feature pool. The optimal activation function is selected from five different types that are sigmoidal (1), sinus (2), hard limit (3), triangular basis (4), radial basis (5). Note that upper bound of neurons in the hidden layer is fixed at 30 for both ELM and RELM models. Also, upper bound of neurons in the context layer is fixed at 25 for RELM. The details of the determined parameters are summarized in Table 3.

2.2. Extreme Learning Machine

ELM can be considered a promising learning algorithm for single-hidden layer feedforward neural networks (SLFNs). ELM is a faster learning algorithm compared to other conventional learning algorithms such as backpropagation algorithm. Furthermore, no parameters are needed to be tuned except predefined network architecture [16]. The general structure of an ELM and the algorithm for training this network can be summarized as follows [17]:

For N arbitrary distinct samples (x_i, t_i), where $x_i=[x_{i1},x_{i2},\ldots,x_{in}]^T \in R^n$ and $t_i= [t_{i1},t_{i2},\ldots,t_{im}]^T \in R^m$, standard SLFNs with $\tilde{N}$ hidden nodes and activation function $g(x)$ are mathematically modeled as (Eq. 4):

$$\sum_{i=1}^{\tilde{N}} \beta_i\, g_i(x_j) = \sum_{i=1}^{\tilde{N}} \beta_i\, g(w_i x_j + b_i) = o_i, \quad j = 1, \ldots, N \tag{4}$$

where $w_i = [w_{i1}, w_{i2}, \ldots, w_{in}]^T$ is the weight vector connecting the i[th] hidden node and the input nodes, $\beta_i = [\beta_{i1}, \beta_{i2}, \ldots, \beta_{im}]^T$ is the weight vector connecting the i[th] hidden node and the output nodes, and b_i is the threshold of the i[th] hidden node. $w_i x_j$ denotes the inner product of w_i and x_j.

Standard SLFNs with $\tilde{N}$ hidden nodes and n number of neuron in input layer with activation function $g(x)$ can approximate these N samples with zero error means that $\sum_{j=1}^{N} \|o_j - t_j\| = 0$.

The above N equations can be written compactly as (Eqs. (5) to (7), respectively):

$$H\beta = T \tag{5}$$

where

$$H(w_1,\ldots,w_{\tilde{N}},b_1,\ldots,b_{\tilde{N}},x_1,\ldots,x_N) = \begin{bmatrix} g(w_1x_1+b_1) \cdots g(w_{\tilde{N}}x_1+b_{\tilde{N}}) \\ \vdots \quad \ddots \quad \vdots \\ g(w_1x_N+b_1) \cdots g(w_{\tilde{N}}x_N+b_{\tilde{N}}) \end{bmatrix}_{N\times\tilde{N}} \tag{6}$$

$$\beta = \begin{bmatrix} \beta_1^T \\ \vdots \\ \beta_{\tilde{N}}^T \end{bmatrix}_{\tilde{N}\times m} \text{ and } T = \begin{bmatrix} t_1^T \\ \vdots \\ t_N^T \end{bmatrix}_{N\times m} \tag{7}$$

The hidden layer output matrix of the neural network is denoted as H. The ith column of H is the ith hidden node output with respect to inputs $x_1, x_2, \ldots, x_N$. T is the output of the neural network as it is obvious from equations. Then, output weight matrix ($\hat{\beta} = H^{\dagger}T$) is calculated. $H^{\dagger}$ is the MP generalized inverse of matrix H. Details can be found in [17].

2.3. Recurrent Extreme Learning Machine

RELM has been shown a novel training approach for a single hidden layer Jordan recurrent neural network whose output can be defined by [12]:

$$T=\sum_{i=1}^{m} \beta_i \, g\left(\sum_{j=1}^{n} w_{i,j}x_j + \sum_{j=n+1}^{n+r} w_{i,j}\delta\,(t-j+n)+b_i\right) \quad (8)$$

where m and n are the neuron numbers in the hidden and input layers, respectively. δ denotes delay, t shows the instance order, r is the employed context neuron numbers, which are backward connections from the output layer to the input layer (Eq. 8). Proposed RELM is seen in Figure 2.

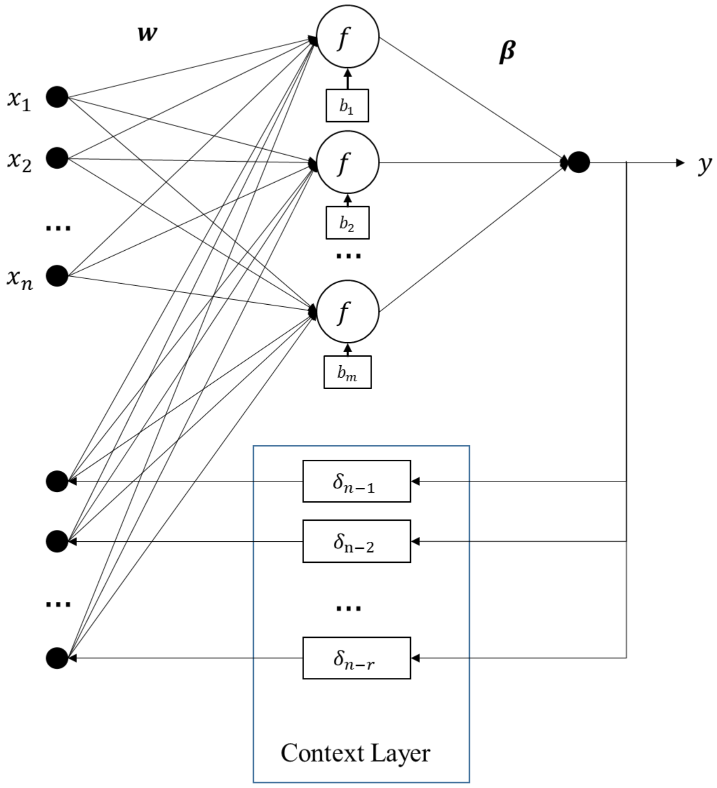

Figure 2. Recurrent extreme learning machine

Context neurons are employed as if they are extra input neurons and hold delayed values of the output. Then, the same learning procedure of ELM is used to determine the output weights. In RELM, the feedbacks are assigned as new inputs with delay and added to the H matrix. Details can be found in [12].

3. RESULT AND DISCUSSION

To evaluate the prediction quality and performance of HS-ELM and HS-RELM models, our work is applied to three stocks listed in BIST50 that are Aksa Energy (AKSEN), Doğan Group of Companies (DOHOL), Şişecam Group (SISE). Training dataset is between 17 April 2013 and 11 September 2015 and there are 1200 observations. Testing dataset is between 14 September 2015 and 30 November 2015 and there are 60 observations.

Five different loss functions namely mean absolute percent error (MAPE), mean absolute error (MAE), mean square error (MSE), root mean square error (RMSE), and Theil's inequality coefficient (TheilU) are utilized to evaluate the performance of HS-ELM and HS-RELM models.

They are calculated as follows (Eqs. 9 to 13 respectively):

$$\text{MAPE}=\frac{100}{\text{N}}\sum_{\text{t}=1}^{\text{N}}\left|\frac{\text{X}_\text{t}\text{-F}_\text{t}}{\text{X}_\text{t}}\right| \tag{9}$$

$$\text{MAE}=\frac{1}{N}\sum_{t=1}^{N}|X_t-F_t| \tag{10}$$

$$\text{MSE}=\frac{1}{\text{N}}\sum_{\text{t}=1}^{\text{N}}(\text{X}_\text{t}\text{-F}_\text{t})^2 \tag{11}$$

$$\text{RMSE}=\sqrt{\frac{1}{\text{N}}\sum_{\text{t}=1}^{\text{N}}(\text{X}_\text{t}\text{-F}_\text{t})^2} \tag{12}$$

$$\text{Theil U}=\frac{\sqrt{\frac{1}{\text{N}}\sum_{\text{t}=1}^{\text{N}}(\text{X}_\text{t}\text{-F}_\text{t})^2}}{\sqrt{\frac{1}{\text{N}}\sum_{\text{t}=1}^{\text{N}}(\text{X}_\text{t})^2}+\sqrt{\frac{1}{\text{N}}\sum_{\text{t}=1}^{\text{N}}(\text{F}_\text{t})^2}} \tag{13}$$

Table 4. The comparison of proposed models

	Model type	MAPE	MAE	MSE	RMSE	TheilU
AKSEN	HS-RELM	4.1769	0.0024	0.000011	0.0033	0.0287
	HS-ELM	13.5789	0.0076	0.000111	0.0105	0.0896
DOHOL	HS-RELM	0.6262	0.0022	0.00001	0.0031	0.0042
	HS-ELM	14.0359	0.0514	0.003465	0.0588	0.085
SISE	HS-RELM	3.5944	0.0017	0.000004	0.0021	0.0218
	HS-ELM	12.7325	0.0058	0.000049	0.007	0.0724

Note that F_t is predicted stock price, X_t is actual stock price and N is total number of tests. Resulting values of performance measures are given in Table 4. Note that, smaller values of MAPE, MAE, MSE, RMSE, and TheilU show better prediction performance for AKSEN, DOHOL, and SISE. Specifically, the MAPE performance of HS-RELM model in DOHOL shows that the prediction error is 0,6262% that is quite reasonable for such a very complex stock prediction model.

4. CONCLUSION

In recent years, stock price prediction is gaining more attention but no consensus exists among researchers as to which type of model is the best for stock price prediction and it is not obvious how to extend this model under different types of stock market. In literature, researchers generally iterate ELM and RELM models many times until they get a satisfying result. To create a more systematic approach for ELM and RELM models, hybrid models that overcome some critical shortcomings of a single model can be used. In this study, we proposed two hybrid methodologies for selecting the most relevant technical indicators and determining the most suitable structure of ELM and RELM, simultaneously. Thus, we are concerned with the design of the hybrid ELM and RELM models by providing a more sensitive and comprehensive setting using HS. Furthermore, we aim to unveil the extent to how HS-ELM and HS-RELM have been used to predict the one-day ahead prediction of daily closing price of stocks listed in BIST50 Index. Based on the results, although the prediction performance of HS-RELM model is significantly better than HS-ELM model, both models are highly promising and can be used in stock price prediction.

5. REFERENCES

1. Zahedi, J., Rounaghi, M.M., 2015. Application of Artificial Neural Network Models and Principal Component Analysis Method in Predicting Stock Prices on Tehran Stock Exchange, Physica A: Statistical Mechanics and its Applications, 438: 178-187.
2. Majumder, M., Hussian, M.D.A., 2007. Forecasting of Indian Stock Market Index Using Artificial Neural Network, Available at: https://nseindia.com/content/research/FinalPaper206.pdf.
3. Wei, L.Y., Cheng, C.H., 2012. A Hybrid Recurrent Neural Networks Model Based on

Synthesis Features to Forecast the Taiwan Stock Market, International Journal of Innovative Computing Information and Control, 8(8): 5559-5571.

4. Zhu, Q.Y., Qin, A.K., Suganthan, P.N., Huang, G.B., 2005. Evolutionary Extreme Learning Machine, Pattern Recognition, 38: 1759-1763.

5. Huang, G., Huang, G.B., Song, S., You, K., 2015. Trends in Extreme Learning Machines: a Review, Neural Networks, 61: 32-48.

6. Bazi, Y., Alajlan, N., Melgani, F., AlHichri, H., Malek, S., Yager, R.R., 2014. Differential Evolution Extreme Learning Machine for the Classification of Hyperspectral Images, Geoscience and Remote Sensing Letters, 11: 1066-1070.

7. Yang, H., Yi, J., Zhao, J., Dong, Z., 2013. Extreme Learning Machine Based Genetic Algorithm and its Application in Power System Economic Dispatch, Neurocomputing, 102: 154-162.

8. Suresh, S., Saraswathi, S., Sundararajan, N., 2010. Performance Enhancement of Extreme Learning Machine for Multi-category Sparse Data Classification Problems, Engineering Applications of Artificial Intelligence, 23: 1149-1157.

9. Hegazy, O., Soliman, O.S., Salam, M.A., 2015. FPA-ELM Model for Stock Market Prediction, International Journal of Advanced Research in Computer Science and Software Engineering, 5: 1050-1063.

10. Wang, F., Zhang, Y., Xiao, H., Kuang, L., Lai, Y., 2015. Enhancing Stock Price Prediction With a Hybrid Approach Based Extreme Learning Machine, 2015 IEEE 15th International Conference on Data Mining Workshops, 1568-1575.

11. Li, X., Xie, H., Wang, R., Cai, Y., Cao, J., Wang, F., Min, H., Deng, X., 2016. Empirical Analysis: Stock Market Prediction Via Extreme Learning Machine, Neural Computing and Applications, 27: 67-78.

12. Ertugrul, Ö.F., 2016. Forecasting Electricity Load by a Novel Recurrent Extreme Learning Machines Approach, Electrical Power and Energy Systems, 78: 429-435.

13. Ólafsson, S., 2006. Metaheuristics, Handbooks in Operations Research and Management Science, 13: 633-654.

14. Saka, M.P., 2009. Optimum Design of Steel Skeleton Structures, In Music-Inspired Harmony Search Algorithm, Springer Berlin Heidelberg, 87-112.

15. Geem, Z.W., 2009. Music-inspired Harmony Search Algorithm Theory and Applications, (Zong Woo Geem ed.). Springer-Verlag Berlin Heidelberg.

16. Wang, Y., Cao, F., Yuan, Y., 2011. A Study on Effectiveness of Extreme Learning Machine, Neurocomputing, 74: 2483-2490.

17. Huang, G.B, Zhu, Q.Y., Siew, C.K., 2006. Extreme Learning Machine: Theory and Applications, Neurocomputing, 70: 489-501.