Professor Sedat METLEK, PhD
E-mail: sedatmetlek@mehmetakif.edu.tr
Department of Economic Informatics and Cybernetics
Burdur Mehmet Akif Ersoy University

FORECASTING OF DOW JONES SUKUK INDEX PRICES USING ARTIFICIAL INTELLIGENCE SYSTEMS

Abstract. *The objective of this paper is to predict the price of the Sukuk index using artificial intelligence models. The Dow Jones Middle East and North Africa (DJMENA) Sukuk indices and the Dow Jones Gulf Cooperation Council (DJGCC) Sukuk indices were chosen as data sets. In addition to the Long Short-Term Memory (LSTM), Recurrent Neural Network (RNN), and Radial Basis Function Neural Network (RBFNN) models in the literature, a new RNN-based artificial neural network model has also been developed. As a result of the research, the MAE values obtained from the DJMENA data set for LSTM, RNN, the proposed model, and RBFNN, respectively, are 0.95, 0.32, 0.22, and 1.16. The MAE values of 1.24, 0.49, 0.20, and 1.58 were obtained for the DJGCC dataset, in the same order. The best predictive performances of the model are (1) the proposed model, (2) RNN, and (3) LSTM, and the lowest performance, (4) RBFNN, respectively. The main contribution of the study is the comparison of RNN, RBF, and LSTM models in the estimation of DJMENA and DJGCC indices and the development of a new RNN-based model that provides the best performance.*

Keywords*: Long Short-Term Memory, Recurrent Neural Network, Radial Basis Function, Artificial Intelligence, Sukuk Index Forecasting.*

JEL Classification: C45, E37

1. Introduction

The enormous increase in global funding needs, the diversity of financial instruments, and the availability of new financial products such as Sukuk have increased the importance of financial forecasts for effective decision making in investment strategies.

Time series forecasting has an extensive study background and is still a hot topic in the literature. Most of the existing researches has focused on increasing and improving the prediction accuracy of neural networks and deep learning. Artificial Neural Network (ANN) and Deep Neural Network (DNN) have significant advantages in terms of modeling nonlinear relationships with high accuracy even if there are extreme values [1], the ability to work with incomplete

DOI: 10.24818/18423264/56.1.22.07

and erroneous data and to work with qualitative data as well as quantitative data [2]. These advantages cause Artificial Intelligence (AI) models to be used more and more day-to-day to forecast financial data.

Previous studies reveal that they have high accuracy in forecasting stock prices [3], stock market indices, exchange rates, cryptocurrencies, and gold prices. Ma, compared Long Short-Term Memory (LSTM), ANN, and Autoregressive Integrated Moving Average (ARIMA) models for stock price forecasting and, as a result, found that LSTM is superior to others, and ANN is better than ARIMA [4]. As similar, Huang et al. combined a genetic algorithm with LSTM for forecasting the financial data and found LSTM was superior to the baseline models[5]. Berradi et al. forecasted the stock prices by combining Recurrent Neural Network (RNN) and Principal Component Analysis (PCA) methods [6]. Chandra and Chand developed ANN-based and RNN-based. mobile application and demonstrated that RNN has better generalization ability compared to ANN for real-world time series problems [7]. Besides, there are algorithms developed based on Radial Basis Function Neural Network (RBFNN) for stock price and trend forecasting [8] or option price validation. Durojaye and Odeyemi proved that RBF model was an adaptable interpolation method to evaluate option prices [9]. Although AI models are being used more and more to predict various financial data, there are still specific topics in finance that have not been studied much using AI. The sukuk, the Islamic capital market instrument, is among these issues.

Securities structured as asset-backed or asset-based in order to meet the financing needs in accordance with Islamic principles in capital markets are called Sukuk [10]. They give the investor undivided ownership right over the underlying assets and, accordingly, the right to receive a share from the income or profit at the rate of participation. Sukuk must comply with the principles of primarily the prohibition of interest, the prohibition of gambling, the prohibition of speculation, the principle of profit-loss-risk sharing, asset-based financing, etc. In recent years, especially socially responsible sukuk and green sukuk have attracted the attention of western countries in long-term environmentally friendly project financing. Sukuk, the newest asset class in the finance industry, the global market reached $ 543.4 billion, which represents 22% in 2.5 trillion dollars of global Islamic financial assets, in 2020 [11].

The strong growth of Islamic finance in recent years has also led academic studies to focus on sukuk. There have been limited studies on the analysis of Sukuk data using AI. These limited eligible studies mostly classify the Sukuk credit risk rating grade [12], [13]. However, studies that forecast Sukuk or Sukuk index prices with AI systems are almost negligible. Çetin, forecasted Sukuk prices with the ANN model by using volatility, dollar, S&P Sukuk indices ,and Eurobond prices as input variables [10]. Hila et al, predicted the Sukuk data by using the outputs of the moving average model as the input of the ANN model [14].

Therefore, to fill the gap in the literature, this study aims to forecast the prices of the S&P Dow Jones Sukuk index with fundamental and advanced AI models. In line with this, two regions were selected to analyze: (i) S&P MENA, dollar-denominated, investment Sukuk issued in the Middle East and North Africa (MENA), and (ii) S&P GCC, dollar-denominated, investment Sukuk issued in the Gulf Cooperation Council (GCC) countries. The Dow Jones Sukuk index is the first published index and is widely used as a benchmark rate of return for investors around the world. Benchmark rate refers that investor consider before decision-making regarding their investments to reducing the risk and uncertainty in the Sukuk markets.

Sukuk index prices are forecasted with four different models as RBFNN, RNN, and LSTM. In addition to these models, a new RNN-based architecture has been proposed to forecast the prices of the Sukuk index. In particular, the proposed RNN-based model ensured that the data at t-1 and t periods of time can be used as input in the internal structure of the proposed architecture. Through this methodology, three important research questions are tried to be answered:

(Q1) What are the determinants of forecasting the Sukuk index?

(Q2) Is there any difference in forecasting performance between artificial intelligence models such RBFNN, RNN, LSTM, and proposed RNN-based model?

(Q3) Do the forecasting performances of the applied models differ regionally?

The contributions of the research are as follows:

- Developing and testing of a new RNN-based proposed model to forecast sukuk index prices.
- Comparison of the success of Sukuk index forecasting with a new deep learning method LSTM and the fundamental AI methods such as RBFNN and RNN.
- Determining the predictors of Sukuk index prices such as the Volatility Index (VIX), the Dollar Index (DXY) and bond prices.

The study is presented in six sections: Section 2 determined the datasets, inputs, and outputs data along with explained the methodology. Section 3, explained the experimental study and the hyper-parameters used in the study. Section 4 presented the findings of the designed models and displayed the error functions used and their results. Section 5 discussed the successful performance of the proposed models and the results of the analysis in the perspective of financial time series and Sukuk markets.

2. Material and Method

The success of an AI algorithm depends on clearly stating the problem and properly modeling the complex nonlinear relationship between input and output variables. In particular, the determination of input variables in the prediction of financial time series with artificial intelligence is of significance, as it directly affects the performance of the model. The features used in the study were determined using the index-based approach. Thus, the power of index values to represent many parameters with a single value was utilized. The preferred index-based approach has two major advantages:

Since the input parameters are reduced, the model is simplified and unnecessary computational load is avoided. As all parameters will affect the index data, data loss is prevented as much as possible. The indices used as input parameters in the study were determined on the basis of economic theories and empirical studies in the literature. Input variables; (i) S&P Bonds, (ii) VIX, (iii) DXY indices, and the output variable is the DJ Sukuk index. The variables used in the study are presented in Table 1.

Table 1. Output and inputs parameters

DATASETS	OUTPUT	INPUTS		
	Y	X1	X2	X3
Dataset 1	DJGCC	S&P GCC Bond	VIX	DXY
Dataset 2	DJMENA	S&P MENA Bond	VIX	DXY

2.1. Description of datasets

In Table 1, while the output is shown with the Y notation, the inputs are shown with the Xn notation. Data that could not be provided due to missing data and differences in vacation dates were removed from the set, and as a result, 1880 daily data between 30 August 2013 and 25 February 2021 were obtained.

First input parameter is the VIX index, which is defined as an indicator of investors' fear. It is designed to measure the market expectation of 30-day volatility by S&P 500 Index option prices. The VIX index published by the CBOE is the most comprehensive and widely used index to represent market risk.

The second input parameter is the DXY index which reflects the value of the dollar against the weighted geometric average of a basket of major currencies (Euro, Japanese Yen, British Pound, Canadian Dollar, Swedish Krona, and Swiss Franc). Since the Dow Jones Sukuk indices dollar-denominated Sukuk issuances, they sustained currency risk. The DXY index is included in the study to represent the exchange risk.

The third input parameter is the bond prices. It is an interest-based and fixed-income instrument that represents a loan made by an investor to a borrower, typically corporate or governmental issuances. There are many studies in the literature that reveal a high relationship between Sukuk and conventional bonds [15], [16]. S&P GCC Bond expresses the bonds issued on Gulf Cooperation Council markets, and the S&P MENA Bond expresses the bonds issued in the Middle East and North Africa countries.

The output parameters of the system are the S&P Dow Jones GCC (DJGCC) Sukuk Index and the S&P Dow Jones MENA (DJMENA) Sukuk Index. S&P Global issues regional Sukuk indices such as DJGCC and DJMENA. DJGCC express dollar-based Sukuk issued in Gulf Cooperation Council markets, DJMENA express dollar-based Sukuk issued in Middle East and North Africa countries.

The time-series graphs of all the variables used in the study are displayed in Figure 1.

Although DJMENA and DJGCC represent Sukuk issued in different regions, there is a strong correlation between them, 99.95%. Dataset subjected to the normalization process and divided into two parts as training (80%) and test (20%) according to the K-fold 5 value. In the next step, the AI algorithms described in Section 3.2 were trained and the testing process was conducted.

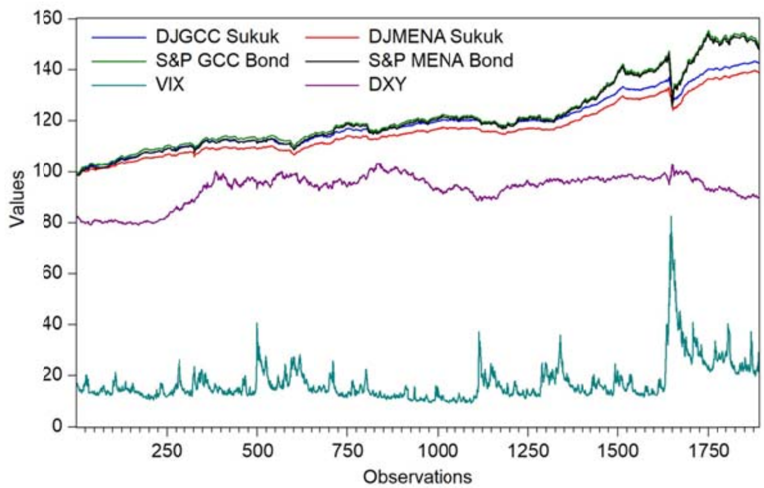


Figure 1. Data graphs

2.2 Method

2.2.1 The Radial Basis Function Neural Network (RBFNN)

Radial Based Function Neural Networks (RBFNNs), first developed by Broomhead and Lowe in 1988 [17], are one of the widely used methods for classification and time series in the literature [18]. RBFNNs generally consist of three layers as input, hidden, and output[19]. The input layer consists of n-dimensional input data entering the network. The hidden layer consists of k Radial Basis Functions (RBF)[9].

The difference of RBFNNs from classical ANN is that radial basis activation functions are used in the transition from the input to the hidden layer. The structure between the hidden and the output layer is the same as in multi-layer ANNs. The training process is also taking place in these hidden layers. The designed structure of the network is illustrated in Figure 2.

Linear, Cubic, Gaussian, Multi-Quadratic, Inverse Multi-Quadratic functions can be used as activation functions in an RBFNN structure. Gaussian function was preferred in the study and its formula presented in Eq. (1).

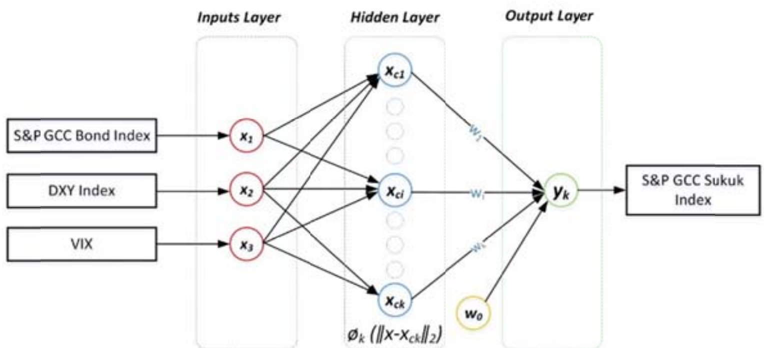


Figure 2. RBFNN structure.

$$Gaussian: \emptyset_k(\,x) = \exp\left(\frac{-\left\|x-c_k\right\|_2{}^2}{2\sigma^2}\right) \tag{1}$$

In Eq. (1), x represents the input vector, c_k refers to the centers, and σ refers to the spread parameter or standard deviation [9]

2.2.2 Recurrent Neural Network (RNN)

Recurrent Neural Networks is a powerful ANN model specially developed for modeling time series sequences. Thanks to this structure, weather forecasting [17], handwriting recognition, and language modeling [20] are used in many different applications [21]. RNNs are also used especially in financial.

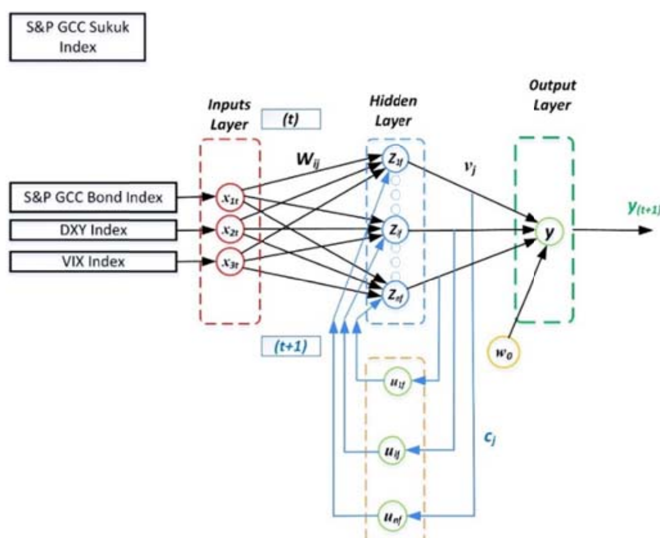


Figure 3. ERNN model

In the designed RNN structures, the layers can be completely connected to each other or partially connected to each other to create different structures. In the RNN structure, although the cells in the hidden layer are part of the feedforward, they can be rejoined as inputs to the computation. Thus, time-delayed feedback is provided from the second node in the hidden layer to the previous node. Due to this structure, it is seen that it gives good results in time series applications in the literature[22].

In the study, the Elman Recurrent Neural Network (ERNN) model with recursive network structure was preferred due to the use of time series type data [23]. Figure 3 shows the structure of a multi-input and single-output ERNN model according to the study. As displayed in Figure 3, in an ERNN model the outputs of the hidden layer are allowed to feedback to themselves through a buffer called the recursive layer.

2.2.3 Long Short Term Memory (LSTM)

Recursive Neural Network (RNN) models are used in many applications for modeling sequential data such as time series. In the RNN method, each data set can be reprocessed with the previous output. Thus, the learning process is carried out considering the data from the previous step. RNNs may be insufficient in consecutive extreme changes in the data during learning.

In order to solve this problem, different a RNN structure has been developed in the literature and the LSTM architecture has been created [24]. LSTM architecture is also recognized as a deep learning method in the literature [25]. Figure 4 displayed the LSTM structure consisting of repeating sequential blocks, as well as a single LSTM cell that generates this architecture. In Figure 4, in step t, c_t denotes the state of the cell, h_t denotes the hidden output state, and x_t denotes the cell's input. The i, f, g and o parameters refers to the input, forget, output gates, and input layer, respectively. We designed and trained the LSTM model separately for the regions in the research. The input and output parameters of the designed structure are presented in Table 1 in detail. To forecast output values from the prepared data set, the hyper-parameters and tested value ranges used in the designed LSTM model are presented in Table 2. Grid Search algorithm was used to determine the LSTM parameters.

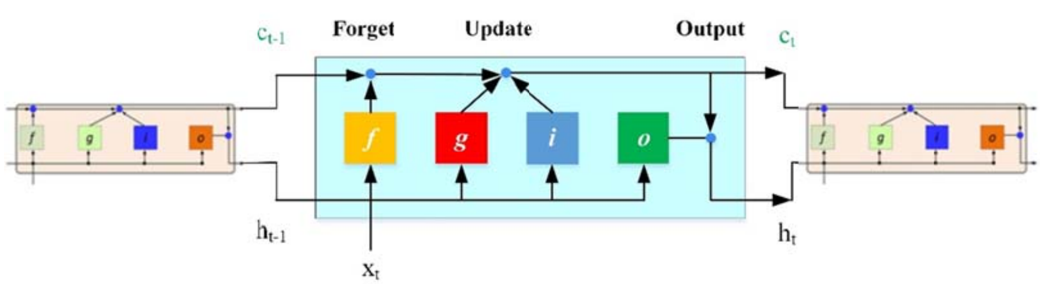


Figure 4. Fundamental structure of LSTM architecture

2.2.4 Proposed RNN model

RNN models are preferred because they produce successful results in the forecasting of the time series. However, fundamental RNN models may be insufficient if the time series is affected by the lag length and contains strong structural fractures. A new RNN-based model was developed to increase prediction performance due to the existence of the conditions mentioned in the datasets used in the study.

In the proposed model, the structure presented in Figure 5 was created in order not to be affected by fractures in the output data. The created structure consists of two phases, forward and back propagation. The forward propagation algorithm of the model consists of six steps, and these steps are explained below.

Step 1: For the x_{ck} cell in Hidden Layer 1, it is first essential to find the total value of the cell. For this, values in the range of 0-1 are randomly assigned to the weights between the cells. Then, in the $t=0^{th}$ iteration, the data coming from the input layer (X_k) and the (R_{c-1}) and (S_{c-1}) values from the *t-1* iteration are multiplied by their respective w_{xk}, w_{uk} and w_{pk} weights. This process is formulated by Eq. (3).

$$netx_{ck}(k) = \sum_{k=1}^{n} w_{xk}X_k + \sum_{k=1}^{n} w_{pk}S_{ck} + \sum_{k=1}^{n} w_{uk}R_{ck} \tag{3}$$

The sum value is subjected to the activation function formulated in Eq. (4). Since there is no value coming from *t-1* in the first iteration, (R_{c-1}) and (S_{c-1}) values are taken as zero. After the second iteration, the calculated values are taken in process.

$$x_{ck}(k) = f_H\big(netx_{ck}(k)\big) \tag{4}$$

If the activation function in the hidden layer is sigmoid; $f_H(x) = \frac{1}{(1+e^{-x})}$.

Step 2: After the values obtained from Hidden layer 1 are multiplied with the relevant weights (w_{hk}) and summed, the bias value (rb_k) is added and transferred to Hidden layer 2. This process is formulated by Eq. (5).

$$netR_{ck}(k) = \sum_{k=1}^{n} w_{hk}X_{ck} + rb_k \tag{5}$$

The values calculated by Eq. (5) are subjected to the activation as in Eq. (6).

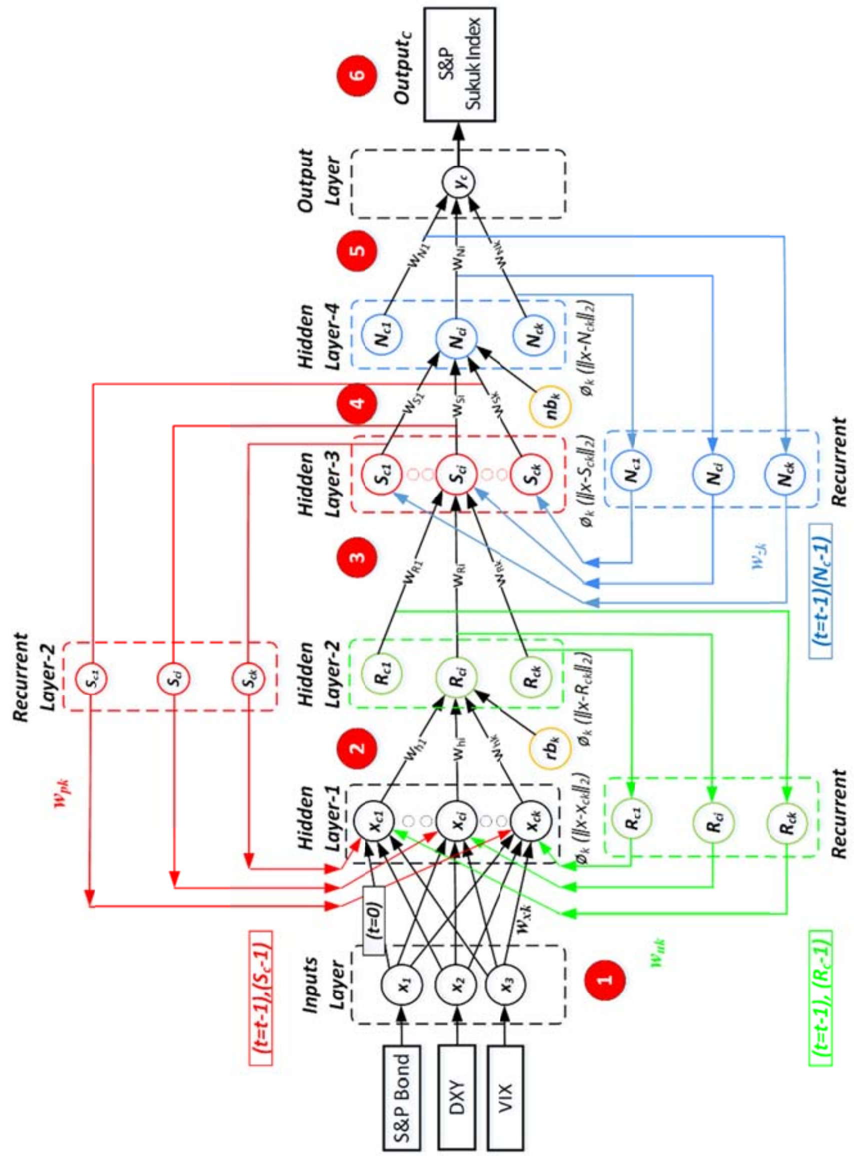
$$R_{ck}(k) = f_H\big(netR_{ck}(k)\big) \tag{6}$$

Step 3: The values acquired from Hidden Layer 2 ($R_{ck}(k)$) and the (N_{ck-1}) values from Recurrent Layer 3 from the t-1^{th} iteration are multiplied by their (w_{Rk}) and (w_{hk}) weights, and Eq. (7) and (8) is implemented.

$$net_{sk}(k) = \sum_{k=1}^{n} w_{Rk}R_{ck} + \sum_{x=1}^{k} w_{zk}N_{ck} \tag{7}$$

$$S_{ck}(k) = f_H\big(net_{sk}(k)\big) \tag{8}$$

Figure 5. Proposed RNN-based model



Step 4: Values in Hidden Layer 3 are included in Step 1 over Recurrent Layer 2. At the same time, the values from Hidden Layer 3 are transferred to Hidden Layer 4 multiplying with the relevant weights *(w_{Sk})*. In Hidden Layer 4, it

is subjected to the sum and activation functions shown in Eq. (9) and (10), respectively.

$$net_{Nk}(k) = \sum_{x=1}^{k} w_{Sk}S_k + wb_k \tag{9}$$

$$N_{ck}(k) = f_H\big(net_{Nk}(k)\big) \tag{10}$$

Step 5: Values in Hidden Layer 4 are included in Step 3 over Recurrent Layer 3. At the same time, Hidden Layer 4 values are multiplied by the relevant weights (w_{Nk}) and transferred to the output layer by subjecting them to the sum function. Afterwards, it is subjected to the activation process one more time. This process is formulated in Eq. (11) and (12).

$$net_{Yk}(k) = \sum_{x=1}^{k} w_{Nk}N_{ck} \tag{11}$$

$$Y(k) = f_H\big(net_{Yk}(k)\big) \tag{12}$$

Step 6: The values of the output layer form the result of the first forward-directed iteration of the entire system. Firstly, the errors between the forecasted and actual values are calculated by the Mean Square Error Function shown in Eq. (13).

$$MSE = \frac{1}{n}\sum_{x=1}^{n}(Y_x - A_x)^2 \tag{13}$$

After the total error value is determined as a result of all training data, the error spreads from back to front as in the MLF-ANN model. In the backpropagation of the error, the weights in the Recurrent 1 and Recurrent 2 layers are not updated, as in the Elman networks. These processes are repeated until the total error reaches the desired level or the specified number of iterations.

During the testing phase of the system, it was ensured that three data lines were used instead of a single data line. If a single line is used here, since there will be no data from S_c, R_c and N_c displayed in Figure 5, these values are taken as zero and as a result, success will decrease. It is important to test the proposed model with at least three rows of data in order to increase the success of time series forecasting and to make a highly accurate prediction.

3. Experimental Study

In the study, four different methods explained in Section 3.2 were tested for each model shown in Table 1. The flowchart of the algorithms is presented in Figure 6. In step 3 of the algorithm, the data is split into two as training and testing according to the K-fold 5 value. After allocated the data for training in this step, hyper-parameters were determined in accordance with the model used in Step 4. Grid search and trial-and-error methods are used, for which the values of hyper-parameters can take an infinite number of values. Value lists were created for the hyper-parameters in the tested ranges using these methods and are presented in Tables (2)-(5).

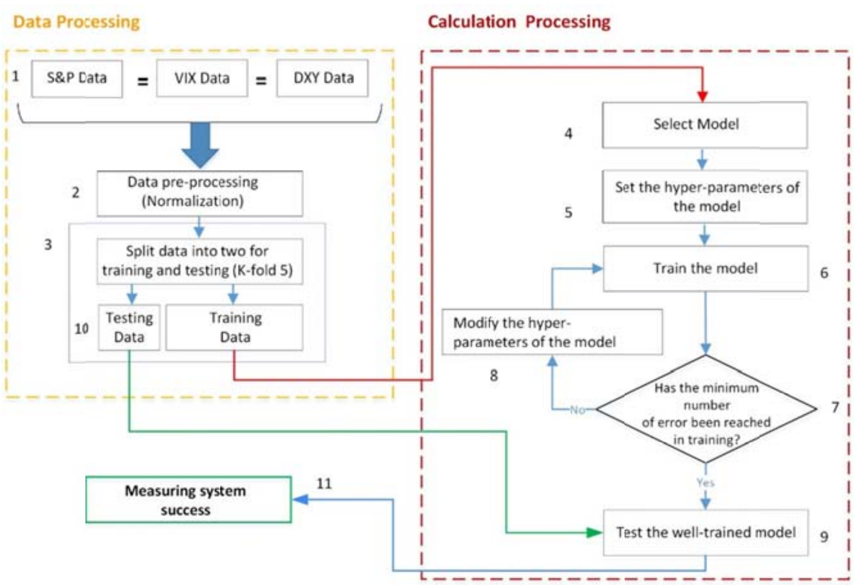


Figure 6. Flowchart diagram

Table 2. LSTM Hyper-parameters

Hyper-parameters	Tested Hyper-parameters Range	Best Values	
		Dataset 1	Dataset 2
Initial learning rate	$10x10^{-4}$-$10x10^{-1}$	$3x10^{-3}$	$7x10^{-3}$
Dropout rate	0.05-0.9	0.17	0.13
Batch size	3-8	6	6
Learn Rate Drop Period	25-250	125	125
Learn Rate Drop Factor	0.05-0.5	0.20	0.17
Max Epoch Iterations	100-800	580	670
Fully Connected Layer	15-200	40	40
Dropout Layer	0.1-0.9	0.55	0.65

Table 3. RBFNN Hyper-parameters

Hyper-parameters	Tested Hyper-parameters Range	Best Values	
		Dataset 1	Dataset 2
Standard Deviation(σ)	0.10-2.00	0.7	0.8
Kernel functions	Linear	Gaussian	Gaussian

	Cubic Exponential Gaussian		
Goal	0.001 – 0.01	0.01	0.01
Number of neurons	3-100	17	25
Max Epoch Iterations	0-1000	897	945

Table 4. Proposed Model Hyper-parameters

Hyper-parameters	Tested Hyper-parameters Range	Best Values	
		Dataset 1	Dataset 2
Hidden Layer 1 Training functions	Levenberg-Marquardt (trainlm) Bayesian Regularization (trainbr) BFGS Quasi-Newton (trainbfg) Scaled Conjugate Gradient (trainscg)	trainbr	trainbr
Hidden Layer 1 neurons	2-20	4	4
Hidden Layer 2 Training functions	Levenberg-Marquardt (trainlm) Bayesian Regularization (trainbr) BFGS Quasi-Newton (trainbfg) Scaled Conjugate Gradient (trainscg)	trainbr	trainbr
Hidden Layer 2 neurons	2-20	3	4
Hidden Layer 3 Training functions	Levenberg-Marquardt (trainlm) Bayesian Regularization (trainbr) BFGS Quasi-Newton (trainbfg) Scaled Conjugate Gradient (trainscg)	trainbr	trainbr
Hidden Layer 3 neurons	2-20	3	3
Goal	0.001 – 0.01	0.01	0.01
Number of neurons	2-20	12	7
Max Epoch Iterations	0-1000	65	72

Table 5. RNN Hyper-parameters

Hyper-parameters	Tested Hyper-parameters Range	Best Values	
		Dataset 1	Dataset 2
Hidden Layer Number of neurons	2-20	10	7
Training functions	Levenberg-Marquardt (trainlm) Bayesian Regularization (trainbr) BFGS Quasi-Newton (trainbfg) Scaled Conjugate Gradient (trainscg)	trainbr	trainbr
Goal	0.001 – 0.01	0.01	0.01
Number of neurons	2-20	12	7
Max Epoch Iterations	0-1000	65	72

4. Results and Discussion

Measuring the success of the model is as important as the model used in any artificial intelligence method. In the study, widely used metrics such as Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and MAPE (Mean Absolute Percentage Error), are used to evaluate the success of the proposed methods. In accordance with the K-fold 5 values, the RBFF, RNN, LSTM, and proposed models defined in detail in Section 3 were tested on the DJGCC and DJMENA dataset, and the success of the systems was calculated.

The performance results of the system based on the error values are presented in Table 6. In Table 6, the lowest forecast performance was acquired with RBFNN in the analysis of all datasets, and as expected, the RNN-based methods produced good results.

Table 6. Success results

Model	Prediction Model	MAE	MSE	RMSE	MAPE
DJGCC (Dataset 1)	RBFNN	1.58417	18.55768	4.30786	0.01215
	RNN	0.49700	0.473258	0.68793	0.00410
	LSTM	1.24628	3.919203	1.97969	0.01052
	Proposed Model	0.20116	0.076672	0.27689	0.00168
DJMENA (Dataset 2)	RBFNN	1.16707	8.090843	2.84444	0.00941
	RNN	0.32703	0.198354	0.44537	0.00276
	LSTM	0.95139	1.552338	1.24592	0.00831
	Proposed Model	0.22591	0.081684	0.28580	0.00193

However, contrary to expectations in the LSTM model, it performed lower than RNN. In the comparison of LSTM and RNN success rates, for DJGCC the MAPE value is 0.01052 with LSTM, while 0.00410 with RNN, and likewise for DJMENA the MAPE value is 0.00831 with LSTM, while 0.00276 with RNN.

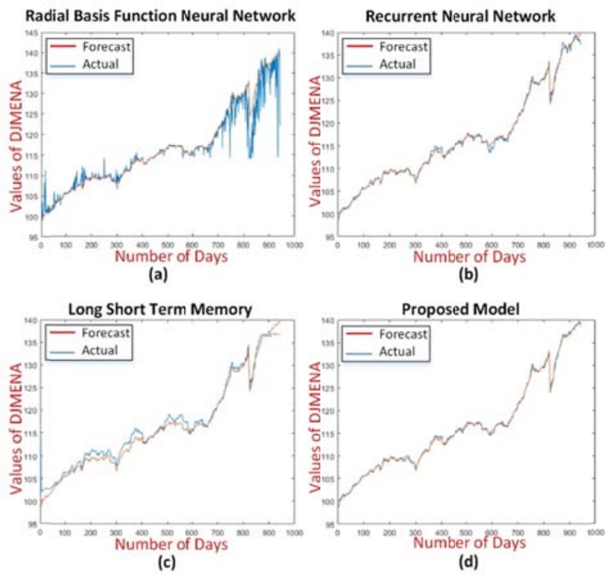


Figure 7. DJGCC forecasting result graphs

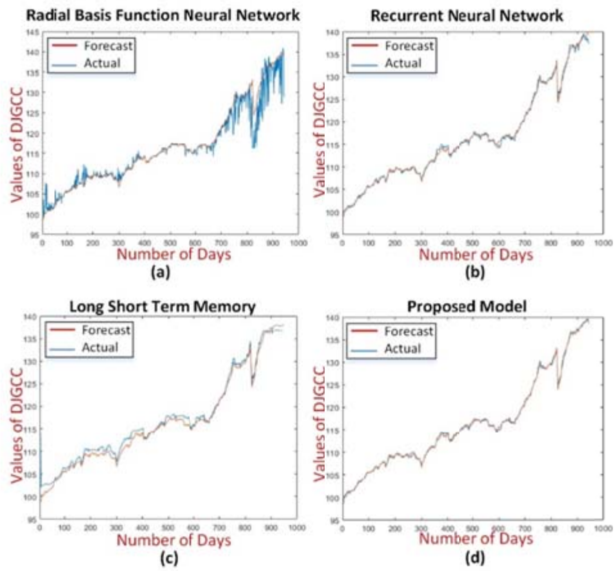


Figure 8. DJMENA forecasting result graphs

In the study, superior success was achieved with the proposed model based on RNN, which obtained MAPE values of 0.00168 and 0.00193 for datasets 1 and 2, respectively. Consistent with these results, it is visualized in Figures 7 and 8 as a graph of the forecast and actual values for the datasets.

5. Conclusion and Recommendation

To forecast the prices of the Sukuk index launched by Standard and Poor's using soft computing methods are tested, four different methods for DJGCC and DJMENA datasets using the hyper-parameters in Tables 3-8 according to the K-fold 5 value. The results of our four experiments give a promising outlook for forecasting Sukuk index prices with soft computing methods. In this part of the study, we can respond to the research questions posed in the introduction section.

(Q1) What are the determinants of forecasting the Sukuk index?

Bond prices, the volatility index, and the dollar index are determined as efficient predictors of the sukuk index forecasting models.

(Q2) Is there any difference in forecasting performance between artificial intelligence models such RBFNN, RNN, and LSTM?

Yes, the models' forecasting performances vary. In the RBFNN method, it is seen that the forecasting performance is low in both datasets. The reason for this may be the existence of extreme values in the VIX index, which is used as an indicator of volatility in financial markets, as input parameters. Therefore, the designed system may disregard the local minimum and maximum points.

The RNN method provides high forecasting performance as a result of being able to model well the autocorrelation link between t and t-1 in the generated dataset. Contrary to previous studies in the literature, lower forecasting performance was acquired in the LSTM method than in the RNN model. The low forecast performance can be attributed to the volatility of the input parameters in the datasets.

The method proposed in the study is a tightly chained RNN-based method, as shown in Figure 7. The relationship between t and t-1 data can be modeled much better due to the autocorrelation of the data and thanks to the sequentially linked RNN structures. Therefore, this proposed structure provided the highest forecasting performance in the study.

(Q3) Do the forecasting performances of the applied models differ regionally?

No significant differences were found between the forecast performance of regional sukuk indices. Since the correlation between the DJGCC and DJMENA datasets is very high, the success performance does not show regional differences.

For future work, the forecasting performance of soft computing methods can be compared with statistical methods such as GARCH, used for financial time series prediction. In addition, new models can be designed by hybridizing statistical and AI methods.

REFERENCES

[1] **M. Kyoung-Sook & K. Hongjoong (2019), *Performance of Deep Learning in Prediction of Stock Market Volatility;*** *Economic Computation and Economic Cybernetics Stdies and Research* 53(2) ; *ASE Publishing;*

[2] **P. Yu & X. Yan (2020), *Stock Price Prediction Based on Deep Neural Networks***; *Neural Comput Appl*, 32(6), 1609–1628;

[3] **S. Donghwan, M. Busogi, A. M. Chung Baek, and N. Kim (2020), *Forecasting Stock Market Index Based on Patterndriven Long Short-Term Memory;*** *Economic Computation and Economic Cybernetics Studies and Research,* 54(3); *ASE Publishing;*

[4] **Q. Ma (2020), *Comparison of ARIMA, ANN and LSTM for Stock Price Prediction,*** *in ISEESE 2020*;

[5] **Y. Huang, Y. Gao, Y. Gan & M. Ye (2020) *A New Financial Data Forecasting Model Using Genetic Algorithm and Long Short-term Memory Network;*** *Neurocomputing*;

[6] **Z. Berradi & M. Lazaar (2019), *Integration of Principal Component Analysis and Recurrent Neural Network to Forecast the Stock Price of Casablanca Stock Exchange,*** *in Procedia Computer Science*, 148, 55–61;

[7] **R. Chandra & S. Chand (2016),** ***Evaluation of Co-evolutionary Neural Network Architectures for Time Series Prediction with Mobile Application in Finance;*** Appl Soft Comput J, 49, 462–473, December;
[8] **D. Wang & Y. Li (2010),** ***A Novel Nonlinear RBF Neural Network Ensemble Model for Financial Time Series Forecasting,*** in 3rd International Workshop on Advanced Computational Intelligence, IWACI 2010;
[9] **M. O. Durojaye & J. K. Odeyemi (2020),** ***Valuation of Option Pricing with Meshless Radial Basis Functions Approximation***; *Asian J Adv Res Reports*, 10(1), 60–70, May;
[10] **D. T. Çetin (2021),** ***Sukuk in the Islamic Finance System***; *First Edit. Ankara, Turkey: Gazi Publishing House*;
[11] **Y. Mnif and M. Tahari (2021),** ***The Effect of Islamic Banks' Specific Corporate Governance Mechanisms on Compliance with AAOIFI Governance Standards;*** *J Islam Account Bus Res*, 12(1), 21–43, January;
[12] **T. Arundina, M. Kartiwi, & M. A. Omar (2016),** ***Artificial Intelligence for Islamic Sukuk Rating Predictions,*** *in Artificial Intelligence in Financial Markets, Palgrave Macmillan UK*, 211–241;
[13] **T. Arundina, M. Azmi Omar & M. Kartiwi (2015),** ***The Predictive Accuracy of Sukuk Ratings; Multinomial Logistic and Neural Network Inferences;*** *Pacific Basin Financ J*, 34, 273–292;
[14] **N. Z. Hila, L. Muhamad Safiih & N. A. Mohamed (2019),** ***Empirical Study of Sukuk Investment Forecasting Using Artificial Neural Network Base Algorithm;*** *Int J Innov Eng Technol*, 13(3), 124–127;
[15] **D. Q. Chen & H. Bin Wang** (2011), ***The Stationarity and Invertibility of a Class of Nonlinear ARMA Models;*** *Sci China Math*, 54(3), 469–478, March;
[16] **L. Ma & S. Tian (2020),** ***A Hybrid CNN-LSTM Model for Aircraft 4D Trajectory Prediction;*** *IEEE Access*, 8, 134668–134680;
[17] **K. G. Liakos, P. Busato, D. Moshou, S. Pearson & D. Bochtis (2018),** ***Machine Learning in Agriculture: A Review;*** *Sensors*, 18(8);
[18] **G. A. Montazer & D. Giveki (2015),** ***An Improved Radial Basis Function Neural Network for Object Image Retrieval;*** *Neurocomputing*, 168, 221–233, Nov.;
[19] **F. Atasoy, B. Sen, F. Nar & I. Bozkurt (2017),** ***Improvement of Radial basis Function Interpolation Performance on Cranial Implant Design;*** *Int J Adv Comput Sci Appl*, 8(8), 83–88;
[20] **Q. Zhang et al. (2020),** ***Transformer Transducer: A Streamable Speech Recognition Model with Transformer Encoders and RNN-T Loss,*** *in ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 7829–7833;
[21] **Y. Bai, J. Xie, C. Liu, Y. Tao, B. Zeng & C. Li (2021),** ***Regression Modeling for Enterprise Electricity Consumption: A Comparison of Recurrent Neural Network and its Variants;*** *Int J Electr Power Energy Syst*, 126, 106612;

[22] **P. Montebruno, R. J. Bennett, H. Smith, & C. van Lieshout (2020),** ***Machine Learning Classification of Entrepreneurs in British Historical Census Data;*** *Inf Process Manag*, 57 (3), 102210;
[23] **O. Cagcag Yolcu, F. Aydın Temel & A. Kuleyin (2021),** ***New Hybrid Predictive Modeling Principles for Ammonium Adsorption: The Combination of Response Surface Methodology with Feed-forward and Elman-Recurrent Neural Networks;*** *J Clean Prod*, 311, 127688;
[24] **C. She, Z. Wang, F. Sun, P. Liu & L. Zhang (2020),** ***Battery Aging Assessment for Real-World Electric Buses Based on Incremental Capacity Analysis and Radial Basis Function Neural Network;*** *IEEE Trans Ind Informatics,* 16(5), 3345–3354, May;
[25] **A. Altan, S. Karasu & E. Zio (2021),** ***A New Hybrid Model for Wind Speed Forecasting Combining Long Short-term Memory Neural Network, Decomposition Methods and Grey Wolf Optimizer;*** *Appl Soft Comput*, 100, 106996.