İşlem Hacmi ve Mevsimsel Değerler Dikkate Alınarak Derin Yapay Sinir Ağı ile Türk Hava Yolları BIST Hisse Fiyatı Tahmini

Araştırma Makalesi/Research Article

Muhammer İLKÇAR

Fethiye Faculty of Business Administration Muğla Sıtkı Koçman University, Muğla, Turkiye
muhammerilkucar@mu.edu.tr
(Geliş/Received:26.09.2022; Kabul/Accepted:13.01.2023)
DOI: 10.17671/gazibtd.1180350

Özet— Bilişim çağının getirdiği veri birikimi, bunların analiz edilerek farklı alanlarda kullanılmasını da beraberinde getirmiştir Veriler, geçmişe dönük bilgi edinme, raporlama, analiz, yapay zekâ ve makine öğrenimi gibi farklı amaçlar için kullanılabilmektedir. Yapay zekâ mühendislik, sağlık, sanayi, üretim, ulaşım, borsa, eğitim, sosyal bilimler gibi farklı disiplinlerde farklı amaçlarla kullanılmaktadır. Bu çalışmada, makine öğrenmesi ile Türk Hava Yolları hisse senedi fiyat tahmini yapılmıştır. Makine öğrenmesi olarak FNN, LSTM ve GRU gibi farklı yapay sinir ağı derin öğrenme yöntemleri kullanılmıştır. Veri seti, Türk Hava Yolları'nın 4 Ocak 2010 ile 31 Ocak 2022 tarihleri arasında BİST' teki günlük borsa endeks bilgilerinden oluşmaktadır. Sistemin eğitimi sırasında olası spekülatif davranışların etkisini azaltmak için işlem hacmi verileri ile birlikte değerlendirilmiştir. Yolcu taşıyan havayollarının gelirleri mevsimsel olarak etkileneceğinden mevsimsel veriler de dikkate alınmıştır. Sistem, farklı kısa-uzun süreli bellek tabanlı yapay sinir ağı modelleri ile eğitilmiş ve test edilmiştir. Modellerin performans göstergeleri olarak R-kare, MSE, RMSE ve MAE kullanılmıştır. Test R-kare performans değerlerine göre sistem FNN' de %97, LSTM ve GRU' da ise %99 başarı göstermiştir. Pandemi nedeniyle aşırı fiyat dalgalanmalarına ve ekonomik krize rağmen yüksek bir performans sergilediği söylenebilir. Bu sonuçlara göre, makine öğrenmesi, sıralı veri seti tahmini için bir karar destek sistemi olarak kullanılabilir. Çalışma ile FNN, LSTM ve türevleri makine öğrenme metotlarının hava yolu taşımacılık sektörü endeks tahmininde başarılı bir şekilde kullanılabileceği sonucuna varılabilir.

***Anahtar Kelimeler*—**ANN, LSTM, GRU, BIST, THYAO, borsa endeks tahmini

Prediction Turkish Airlines BIST Stock Price Through Deep Artificial Neural Network Considering Transaction Volume and Seasonal Values

Abstract— The collection of data in the information age has led to its analysis and use in different fields. Data can be used for different purposes, such as historical information, reporting, analysis, artificial intelligence, and machine learning. Artificial intelligence is used for different purposes in different disciplines such as engineering, health, industry, production, transportation, the stock market, education, and the social sciences. In this study, Turkish Airlines' stock price prediction was made using machine learning. Different artificial neural network methods were used, such as an FNN, LSTM, and GRU. The data set consists of daily stock market index information for Turkish Airlines in BIST between the dates of January 4, 2010, and January 31, 2022. During the training of the system, it was assessed together with the transaction volume data to reduce the effect of possible speculative behavior. Since the income of airlines carrying passengers is seasonally affected, seasonal data are also considered. The system has been trained and tested with different short-long term memory-based artificial neural network models. The performance indicators of the models were used as R-Square MSE, RMSE, and MAE. According to the R-Square, performance score of the test, the success rate of system was 97% in FNN, and 99% in LSTM and GRU. It performed well despite extreme price fluctuations due to the pandemic and economic crisis. According to these results, machine learning can be used as a decision support system for sequential data set prediction. In this study, it can be concluded that FNN, LSTM, and its derivative machine learning methods can be successfully used in air transport sector index prediction.

***Keywords*—**ANN, LSTM, GRU, BIST, THYAO, stock index predict

1. INTRODUCTION

Today, with digitalization, users, electronic devices, sensors, and computer software constantly generates data, which is stored on digital media. This collection of data has led to the creation and development of many new disciplines such as big data, data science, data analysis, machine learning, and artificial intelligence. By organizing the data and using artificial intelligence methods, they are used for many processes such as determining consumer behavior, fraud detection, trends, investment advice, text analysis, anomaly detection, forecasting, classification, clustering, pattern recognition, detecting possible relationships, customer analysis, marketing, sentiment analysis, translation from language to language, smart assistants, etc. Therefore, data has become a strategic weapon today. Data has led the way for companies investing in data science and artificial intelligence to become the world's largest and most competitive businesses Johnston [1]. Data can be of different types and structures depending on the source they are obtained from. The data are grouped as graphs and sequential Tan, Steinbach, and Kumar [2]. Sequential data consist of operations performed by humans or machines. Sequential data are arranged according to a certain quality and order. For example, the stock market index is data arranged according to time, and genome data is data arranged according to a certain rule. Time-dependent data has a temporal dimension, and it should be considered a temporal chronology. This type of data is called a time series or sequential data. Time series can be used in processes such as creating an institutional memory, accessing historical information, reporting, noticing trends, completing missing data, detecting extraordinary situations, and making predictions. A time series of data can consist of components such as trends, seasonal or situational fluctuations, and random movements Duru [3]. Forecasting by using time series can be done with the help of statistical methods or machine learning. As a statistical method for time series prediction methods such as the exponential moving average (EMA: Exponential Moving Average) Hansun [4], non-stationary linear stochastic models (ARIMA: autoregressive integrated moving average) Hyndman and Athanasopoulos, [5] are often used.

Torres et al. [6] studied the use, advantages, disadvantages, and the role of hyperparameters in the time series of popular deep machine- learning algorithms.

Kaynar and Taştan [7] compared the monthly and daily exchange rate data of the T.R. Central Bank with the ARIMA statistical method and ANN.

Bayraktar and Badur [8] predict the index value of the Istanbul Stock Exchange (ISE) by using an artificial neural network, taking into account the index, exchange rate, and overnight interest data.

Aygören, Sarıtaş, and Moralı [9] made the same predictions with traditional methods (ARMA and Newton) and ANN uses approximately 4000 data points belonging to instruments such as the ISE 100 index, gold prices, exchange rates, interest rates, and the results were compared. They showed that ANN obtained more successful results than traditional methods.

Sami, Tavakoli, and Namin [10] made index predictions using ARIMA and LSTM methods using different financial time series data such as the Nikkei 225 Index, the NASDAQ Composite Index (IXIC), the Hang Seng Index (HIS), and the Dow Jones Index (DJ). The performances of these two methods were compared. According to the results they obtained, they showed that the LSTM method was more successful than ARIMA.

Yücesan [11] made a price prediction in the white goods sector using methods such as ARIMA and artificial neural networks (ANN), taking into account different parameters such as the producer price index, house sales, and an exchange rate.

Kalyoncu et al. [12] made predictions by using LSTM as the learning algorithm of the lesson and by using the 2014-2019 indexes of Akbank, Arçelik, Aselsan, Garanti, and Turkish Airlines companies, according to the BIST 30 and stated that the stock market prediction can be made successfully.

Çınaroğlu and Avcı [13] used the dollar rate, oil price, BIST 100 index, and BIST 100 transportation index data as inputs and estimated the THY market value by using the ANN machine learning method.

Güleryüz, Özden, and Gülhan [14] estimated the BIST 30 index with ARIMA and LSTM and stated that LSTM was 26% more successful than the ARIMA method.

Ranjan, Kayal, and Saraf [15] made bitcoin price predictions using the daily price and daily high price frequency information by using Logistic Regression and XBoost machine learning methods. Daily prediction accuracy with Logistic Regression accuracy is %64.8 and XBoost, 5-minute interval prediction accuracy is %59.4.

Solgi, Lo´aiciga, and Kram [16] used the data obtained from past groundwater levels to make groundwater level predictions with the help of LSTM and NN with different scenarios from one day to three months. In their study, they showed the superiority of LSTM over simple NN machine learning methods in water level estimation based on groundwater data only.

Lindemann et al. [17] have made an overview of LSTM and its derivative machine learning methods. They have compared of these methods in nonlinear time series prediction.

Demirel, Cam, and Unlu [18] predicted the BIST 100 index with MLP, SVM, and LSTM methods. They showed that ANN-based methods are more advantageous than classical methods. In their study, THYAO index prediction found the best performance to be 97% in the LSTM method, according to the R-square value.

As we have seen in the literature review, ANN and LSTM machine learning methods are used successfully in time series in different sectors. In this study predicted the stock price of Turkish Airlines (THAYO) using several artificial neural network (ANN) machine learning techniques. THYAO is among the world's best airlines and is publicly traded on the Borsa Istanbul (BIST). The daily average trading volume is 84 million [19]. Consequently, it is selected for this study. As used in the data set, THYAO's daily transaction values were recorded in BIST between January 4, 2010, and January 31, 2022. In the data set, we also considered seasonal effects and transaction volume.

2. ARTIFICIAL NEURAL NETWORK

ANN is a machine learning algorithm developed by imitating the biological learning model of the brain Zou, Han, So; Yang and Yang [20,21]. Especially in recent years, it has shown great development with its success in data processing. It has become one of the most preferred and successful algorithms in artificial intelligence and machine learning due to its success in solving many problems such as image processing, text processing, prediction, classification, and clustering. Therefore, ANN is used for different purposes in different disciplines such as health, engineering, social sciences, industry, and education Zarzycki and Lawrynczuk; Er and Işık [22,23]. An ANN network architecture can be seen in Figure 1, and an ANN perceptron (node) can be seen in Figure 2.

In this study different types of ANN models were used, such as feed-forward backpropagation neural networks (FNN), long-short-term memory (LSTM) units, and gated recurrent units (GRU).

2. 1. ANN Architecture

An ANN architecture is shown in Figure 1. According to figure, an ANN consists of one input layer, one or more hidden layers, and one output layer. The input layer is the layer where data is entered from external sources. Since the input layer is the layer where the data is accepted, the number of nodes is equal to the number of attributes in the input data set. The hidden layer prepares data for the next layer by passing the information coming to its input through the mathematical function as in equation 1. Output layering exporting the calculated values (Equation 1). The number of output layer nodes is arranged in a way that can best express the output value of the network. For example, for a price prediction will be made in the study, a single node is enough in the output layer. All layers are fully connected. These links have weight values (w_i). These parameters pertain to the learning of artificial neural networks. The ANN training is the process of optimizing the network weight values that give the appropriate output against the input data. Although there is no rule about the number of hidden layer nodes, initially it can be determined to be 1.5 times the number of input layer nodes. However, different hidden layer node number architectures should be created and the training and test performances should be compared and the ANN architecture that will give the best performance should be tried to be found.

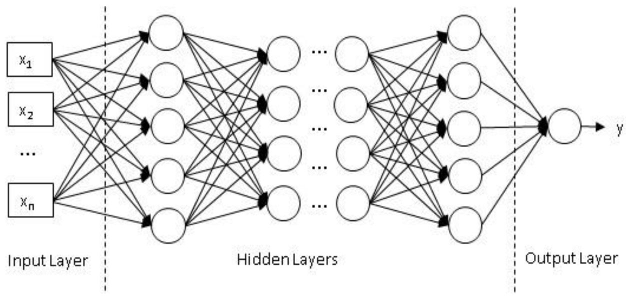

Figure 1. Fundamental three layers an ANN architecture.

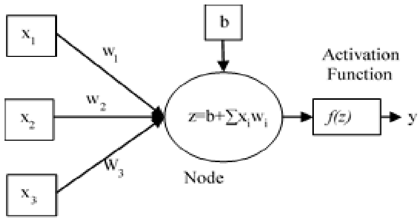

Figure 2. An ANN perceptron (node) structure.

Figure 2 shows the ANN perceptron structure. The perceptron collects the product of its input data (x_i), and the weights (w_i) and obtained a value (z) pass through an activation function (*f(z)*) to produce an output (Equation 1). The b value is the threshold value in the Figure 2 and equation (1). The threshold value ensures that the system works above a certain threshold and is used as an input data attribute.

$$y = f\left(b + \sum x_i w_i\right) \tag{1}$$

The activation function (*f(z)*) ensures that the node output is drawn to a certain range. Thus, the negative effect of excessive data on the performance of the network is minimized by suppressing other data. At the same time, weak data are eliminated if the activation function is below a certain threshold, preventing it from affecting the system. Generally, sigmoid, tangent hyperbolic (tanh), and rectified linear unit (ReLU) functions are preferred as activation functions in the literature because they are easy to derive and calculate (Figure 3. a, b, and c). The sigmoid activation function (Figure 3.a) compresses the perceptron output to the [0, 1] range and the tangent hyperbolic function (Figure 3.b) to the [-1,-1] range. ReLU function (Figure 3.c); If z<=0 for the node sum (z), the output value is 0; otherwise, the output value is z.

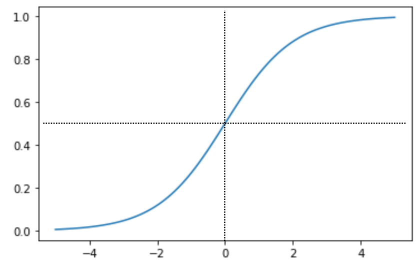

Figure 3.a. Sigmoid

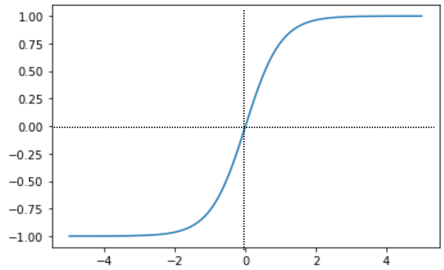

Figure 3.b.Tangent hyperbolic

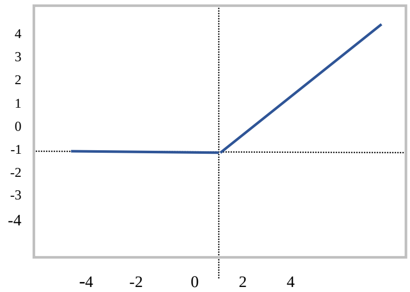

Figure 3.c. ReLU.
Figure 3. a, b, c. Activation functions

2.2. Recurrent Memory-Based Artificial Neural Network

Figure 4 shows the recurrent artificial neural network (RNN) structure. As seen in the figure, a node output is given as an input to itself, thus creating a short-term memory, and it is possible to evaluate the data at time *t* together with the data at the previous (*t-1*) time. When RNN has short-term memory, although it shows a certain success in sequential operations such as text processing, linguistic translation, and time series, it is insufficient. Hochreiter and Schmidhuber [24] developed the RNN to overcome the inadequacies of the RNN node in sequential operations and created a long-short-term memory (LSTM) unit. The LSTM unit enables the previous and subsequent data to be evaluated together by performing operations such as storing and transferring the data for a longer period of time. It is predicted since it is more successful with data sets in sequence, such as text processing and time series Le et al. [25], and Tanışman et al. [26].

An LSTM unit is shown in Figure 5. The LSTM unit consists of four parts; forget, input, state, and output. In this figure, forget gate, data input gate, state gate, and LSTM output gate is in four ANN structures with two layers each. These ANN inputs, $h_{(t-1)}$ and x_t, are combined into a single data set in the form of a heap and form the ANN network input data. The network outputs are in the form of sigmoid and tangent hyperbolic activation functions.

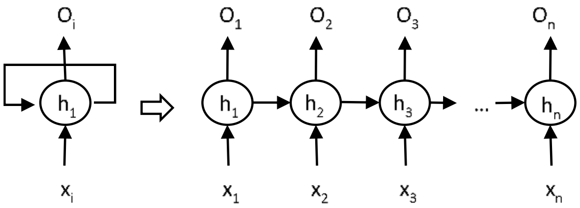

Figure 4. An RNN unit structure

S : Sigmoid function,
t_h : Tangent hyperbolic function,
x_t : Data input,
h_t: LSTM unit output value/short-term memory of network,
$h_{(t-1)}$: Previous LSTM unit output information short-term memory,
$C_{(t-1)}$: Previous state (long-term memory),
C_t : Long-term memory for the next cell,
(.) : dot refers to multiplication as a matrix.

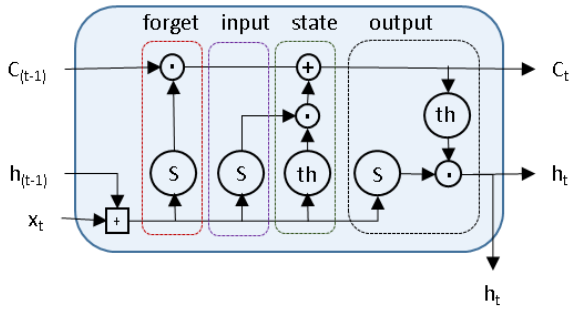

Figure 5. An LSTM unit

Forget Gate (f_t): With the previous output ($h_{(t-1)}$). The input data (x_t) from the forget gate becomes the input to the ANN, and the output layer of the ANN produces a result in the range of [0-1] with the help of the sigmoid activation function (f_t) (Equation 2). This result is multiplied by the value of the previous state ($C_{(t-1)}$) as a dot product (.). If the forget gate output value is 0, the result will be 0, and the $C_{(t-1)}$ state will be forgotten; otherwise, the forget gate output is 1; the $C_{(t-1)}$ state will remain.

Input & state gate (I_t, $\check{C}_t$): These two gates decide whether to update the status value $C_{(t-1)}$. Since the input gate output is a sigmoid function, it produces a result in the range of [0-1]. This value is multiplied by the state gate output, and the new state (C_t) is obtained by adding up to the $C_{(t-1)}$ value. If the input gate output value is 0, the result of the multiplication operation will be 0, so the status will not be updated. If 1, then the state gate output value and $C_{(t-1)}$ will be added together to form the new state (Equations 3, 4, and 5).

Output gate (O_t): x_t and $h_{(t-1)}$ data together are given as input to an ANN whose output activation function is sigmoid. The result obtained is multiplied by the result obtained by passing the C_t value through the tangent hyperbolic function; this will be the unit output (h_t) (Equations 5, 6, and 7).

$f_t = Sig(W_f . [X_t , h_{(t-1)}] + b_f)$ (2)
$I_t = Sig(W_f . [X_t, h_{(t-1)}] + b_i)$ (3)
$\check{C}_t = Tanh(W_c . [X_t, h_{(t-1)}] + b_c)$ (4)
$C_t = f_t . C_{(t-1)} + I_t . \check{C}_t$ (5)
$O_t = Sig(W_o . [X_t , h_{(t-1)}] + b_o)$ (6)
$h_t = O_t . Tanh(C_t)$ (7)

Sig : Sigmoid activation function
Tanh : Tangent hyperbolic activation function
b_i : Threshold

Gated Recurrent Unit (GRU) is another variant of RNN. Its developed by Cho et al. [27, 28]. The GRU architecture has short-term memory like LSTM (Figure 6). As can be seen in the figure, it has a simpler structure than an LSTM unit. Therefore, since it will contain fewer parameters, it will use fewer resources than the LSTM unit. The GRU has a reset port (R) and a status-update port (U), which determine the short-term memory. In Figure 6, the previous state/memory $h_{(t-1)}$ and x_t data are passed through an ANN network with an activation function sigmoid (S) to produce a value in the range of [0-1] (R_t) (Equation 8). In the update gate (U_t), just like in the reset gate, $h_{(t-1)}$ and x_t data are passed through an ANN network whose output is sigmoid (S) and generates a value in the range of [0-1] (Equation 9). The candidate ($\tilde{h}_t$) output is produced by passing the x_t data in equation (10) through the tangent hyperbolic ANN, together with the result of the multiplication of [0-1] with R_t and $h_{(t-1)}$ (Equation 10). For the new long-term memory (h_t) update; ($1-U_t$) is multiplied by $h_{(t-1)}$ plus the product of U_t and $\tilde{h}_t$ (Equation 11).

$R_t = Sig(W_r [h_{(t-1)} + X_t] + b_r)$ (8)
$U_t= Sig(W_u [h_{(t-1)}+ X_t] + b_u)$ (9)
$\tilde{h}_t =Tanh(W_h[R_t h_{(t-1)} + X_t] + b_h)$ (10)
$h_t =(1-U_t).h_{(t-1)} t + \tilde{h}_t . U_t$ (11)

h_t : Long-term memory
W_r : Rest port ANN weight values
W_u : Update port ANN weight values
W_h : Output port ANN weight values
R_t : Reset port
U_t : Update port outcome
$\tilde{h}_t$: Candidate memory

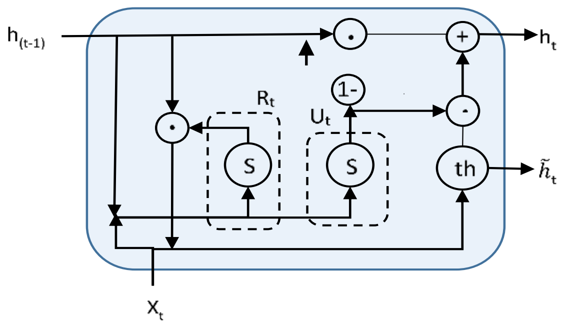

Figure 6. A GRU unit

3. THYAO BIST DATA SET

THYAO's 3036 daily BIST transaction data between January 4, 2010, and January 31, 2022, were obtained from https://tr.investing.com THYAO [19] in .csv (Comma Separated Values) format [29]. As seen in Table 1, the raw data set has transaction date, closing, opening, highest, lowest index, and trading volume indices.

The average price index was calculated by taking the arithmetic average of the opening and closing indexes. The price index change graph by date can be seen in Figure 7. The raw data were transformed into basic data, which consists of price, volume, and season values (Table 1). Season information is also obtained from the transaction date. Attention should be paid to the temporal sequence in time series. It is not appropriate to sort the data according to any other feature other than the time dimension. As seen in Figures 7, and 8 training and test data in time series should not be mixed. It should be separated in a certain order. A certain number of data can be separated from the beginning as training data and the rest as test data. Supervised machine learning methods were used for the prediction of time series. The data set consists of attributes and label values (Table 2).

A supervised learning method was used for the prediction of time series with machine learning. For this purpose, the sequential dataset must be transformed into the supervised machine learning data set. After transformation, the data set consists of attribute values and label values in supervised machine learning (Table 2). Table 2 shows the example transformation for 10 window range. In the table, the window range defines the attributes (x_i), while the first data after the window range represents the label. The volume value was also added as an attribute (x_{11}) in the transformed data set. This attribute consists of the arithmetic means of the volume data. The seasonal effect was added to the transformed data set as an x_{12}. Thus, the dataset will consist of 12 attributes (x_1,x_2,…,x_{12} and one label (y).

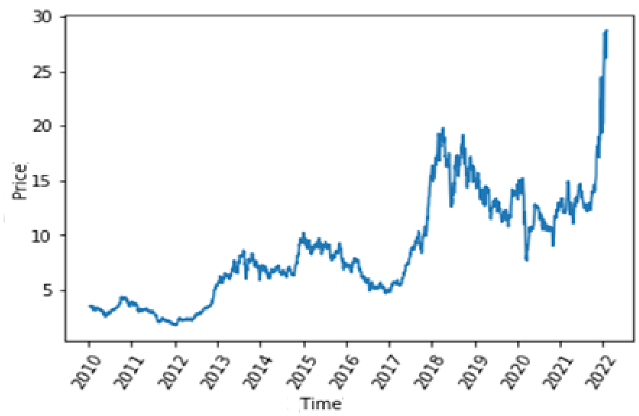

Figure 7. Price graphic of THYAO between 4th of January 2010 and 31th of January 2022

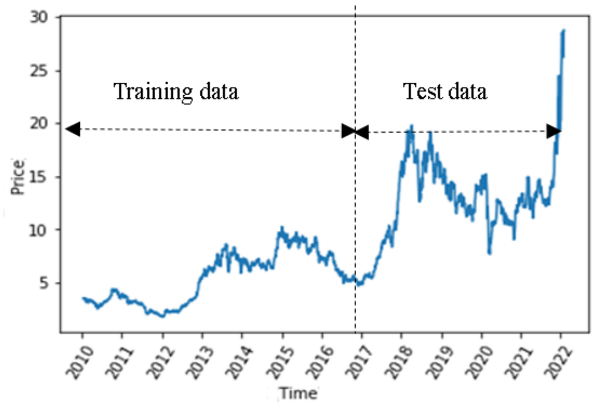

Figure 8. Divided data as training and testing

Table 1. THYAO daily transactions data, volume, and the calculating transaction price

Date	Closed	Open	Highest	Lowest	Volume		Price	Volume	Season
2010.01.18	3,49	3,46	3,49	3,43	5.,80M	→	3.48	5.,80M	1
2010.01.19	3,56	3,52	3,56	3,46	11,50M	→	3.54	11,50M	1
2010.01.20	3,46	3,52	3,52	3,43	16,12M	→	3.49	16,12M	1
2010.01.21	3,46	3,48	3,49	3,40	13,09M	→	3.46	13,09M	1
2010.01.22	3,40	3,43	3,46	3,34	10,54M	→	3.42	10,54M	1
...	...	…	...	...	...	Transforming	....	....	...
2010.01.28	3,46	3,46	3,49	3,40	8.66M	→	3.46	8.66M	1
2022.01.29	3,43	3,43	3,46	3,40	7,61M	→	3.43	7,61M	1
2022.02.01	3,49	3,43	3,49	3,40	7.35M	→	3.46	13.99M	1
2022.02.02	3,46	3,49	3,49	3,40	8,23M	→	3.48	8,23M	1

Table 2. Transformation of time series into supervised machine learning data set

Daily transaction dataset					Transforming to a supervised learning data set							
						Price				Volume	Season	Label
Date	Price	Volume	Season		Date	X_1	X_2	...	X_{10}	X_{11}	X_{12}	Y
2010.01.18	3.48	5.,80M	1		2010.01.08	3.48	3.54	...	3.43	10.12	1	3.46
2010.01.19	3.54	11,50M	1		2010.01.09	3.54	3.49	...	3.46	10.31	1	3.48
2010.01.20	3.49	16,12M	1		2010.01.20	3.49	3.46	...	3.48	9.98	1	3.48
2010.01.21	3.46	13,09M	1		2010.01.21	3.46	3.42	...	3.48	9.18	1	3.43
2010.01.22	3.42	10,54M	1		2010.01.22	3.42	3.39	...	3.43	10.43	1	3.29
...	....	....	...		...		...			...		...
2010.01.28	3.46	8.66M	1		2010.01.28	3.46	3.43	...	3.20	12.15	1	3.21
2022.01.29	3.43	7,61M	1		2022.01.29	3.43	3.46	...	3.21	12.14	1	3.23
2022.02.01	3.46	13.99M	1		2022.02.01	3.46	3.48	...	3.23	12.79	1	3.31
2022.02.02	3.48	8,23M	1		2022.02.02	3.48	3.48	...	3.31	13.56	1.	3.39

Some statistical information about the data is given in Table 3. According to the table, the count of dataset samples is 3036, the standard deviation is 4.81, the arithmetic mean is 8.61, the minimum value is 1.74, and the maximum value is 28.76. Since the skew value is between 0.5 and +1 (0.72) the distribution is moderately skewed, and the kurtosis coefficient is positive (0.33), the probability distribution is leptokurtic.

Table 3. Data set statistical values

Statistical Item	Value
Data count	3036
Mean	8.61
Standard deviation	4.82
Min. value	1.74
Max. value	28.76
Skewness	0.72
Kurtosis	0.33

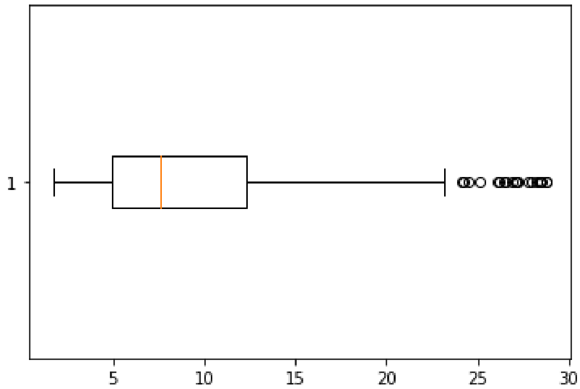

Figure 9: The box graph of the dataset

Figure 9 shows the box plot of the distribution of the data set. According to the graph, the density of the data is between 4 and 13. As can be seen in Figures 7, and 8, while it followed a low profile up and down until 2018 (in 8 years), it showed a rapid increase over 2018, especially in 2022.

4. THYAO STOCK PRICE PREDICTION

In this study, training and prediction processes were carried out by the ANN machine learning algorithm by taking the time series of THYAO's index data in BIST for the last 12 years. In addition to the daily stock market average price index, daily trading volume and seasonal information are also taken into account. For this purpose, the arithmetic average of the trading volume data in the window range and the seasonal information were also added to the data set features. The machine learning process flow chart of the system is given in Figure 10.

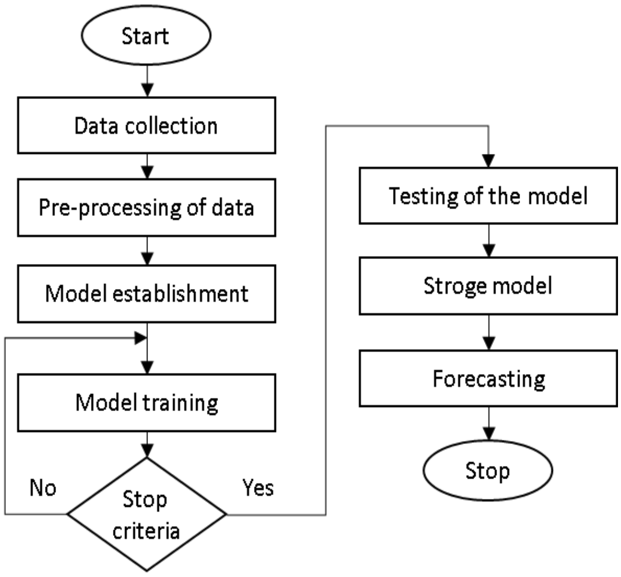

Figure 10. System flow chart

According to Figure 10;
First step; The process starts with data gathering. THYAO daily stock market index data between 04 January 2010 and 31 January 2022 was retrieved from tr.investing.com in csv (comma separated values) format.

The second step is data pre-processing. At this stage, completing and/or deleting missing data in the data set, transforming categorical data, and removing unnecessary features were conducted. As the preprocess in this study:

- A new price feature was created from the arithmetic average of the daily opening and closing indexes of the time series raw data set.
- Transaction volume and season information were added to the data set as a new feature, taking into account the window range.
- By preserving the time dimension of the data, the first 70% of the data was reserved for training, and the remaining 30% was reserved for test data.
- The training and test datasets are shown in Figure 8 supervised machine learning dataset transformations were performed in the form of the attribute (x_i) and label (y) (Table 2).

The third step is the setup of the model. In this step, the selection of the model and its parameters were determined. For example, regarding the ANN model, hyperparameters such as hidden and output layers, the number of nodes, activation functions, loss function, optimization algorithms, learning rate, batch size, the number of iterations, etc. were determined. The model used in the study and the parameters of the model are given in Tables 3, 4, and 5.

In the fourth step; the model was trained using the training data set until the stop criteria were met. The stopping criterion was taken as the point at which the training and validation error values began to differ.

Fifth step; the model was tested with the test data set. In the test process; The model is given a test dataset and is to make a prediction. The difference between the prediction and the actual data shows the amount of error. The average of all test data set error amounts shows the test performance of the system. This value is closer to zero the better the performance of the system. If the test performance is very low or there is a large difference between the training performance and the test performance, there may be cases of memorization or poor learning. For this, all processes in the workflow should be reviewed from the first step, and if necessary, the workflow should be operated again by making corrections.

In the study, the mean of the squared sum of the errors (Equation 12) and the square root of the mean of the squared sum of the errors (Equation 13), the mean absolute error (Equation 14), and the R-square scale (Equation 15) were used as training and tested performance score, and the results were shown in tables 4,5 and 6.

$$\text{MSE} = \frac{1}{n}\sum(z_i - y_i)^2 \quad (12)$$

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum(z_i - y_i)^2} \quad (13)$$

$$MAE = \frac{1}{n}\sum|z_i - y_i| \quad (14)$$

$$R^2 = 1 - \frac{\sum(z_i - y_i)^2}{\sum(z_i - \bar{y}i)^2} \quad (15)$$

y_i : Actual value
z_i : Value predicted by the system
$\bar{y}$: Average actual value
n : Number of samples

After successfully testing the model, it can be stored for later use. Estimation can be done by using a model that has been successful in training and testing. The estimation information obtained assists in decision-making processes. Many factors affect the price index in the real world. Therefore, the prediction information to be obtained from the model can only be used to provide an idea in the decision-making process. This cannot be considered a recommendation. The steps up to the model- building stage in the flowchart are valid for all different models. In this study, training and test performance results of the RNN, LSTM, and GRU models were compared.

4. 1. Feed Forward and Back Propagation FNN Model

FNN is a basic feedforward, backpropagation network model consisting of input, hidden, and output layers. In the study, while making supervised data sets the window interval was taken as 10, and with volume and seasonal data, the total number of features is 12 (Table 2). Therefore, the number of FNN input layer nodes is 12. A single perceptron is sufficient in the output layer, as the model makes price estimations. A single hidden layer with different perceptron numbers was used in the study.

Table 4. ANN hyperparameters

Parameter	*Value*
Batch size	30
Learning algorithm	adam
Delta	0.001
Loss function	MSE
Hidden layers transfer function	ReLU
Output layer transfer function	ReLU

The FNN model hyperparameters are shown in Table 4. The model runs at 100 epochs and doesn't use dropout. The model was run with a different FNN architecture; its effects on the performance of the system were observed, and the obtained data were transferred to Table 5. The table includes the training and test performances. As can be seen from Table 5, performances in all FNN architectures are high and close to each other. If evaluated according to test performance, the lowest MSE error amount was 0.317 and the highest R-square score was 0.966 in the 2nd model, 12-6-1 architecture. Figures 11, and 12 demonstrate these results. Figure 11 shows that the amount of error decreased as the training progressed. In Figure 11 we see that the training and validation error rates decreased together. This indicates that normal learning takes place. In Figure 12, the data predicted by the trained model as a result of the test and the actual data it should have been shown in the same graph. It is seen that they are very close to each other and almost overlap. It represents the high test performance of this model.

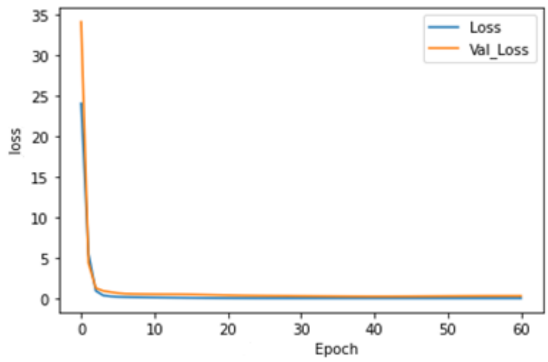

Figure 11. FNN Model Training error change graph

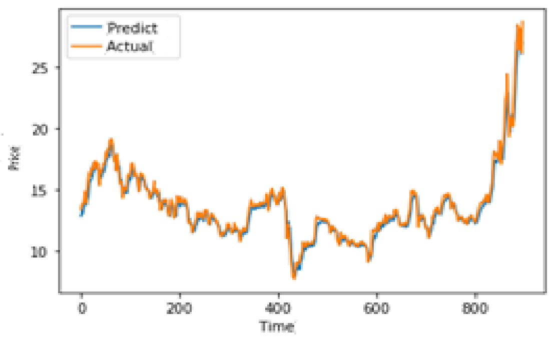

Figure 12. FNN actual value and predicted value.

Table 5. FNN model architecture, optimization parameters, training, and test performance

#	Network Architecture	Training Performance				Test Performance			
	Input-Hidden-Output	MSE	RMSE	MAE	R^2	MSE	RMSE	MAE	R^2
1	12-3-1	0.042	0.204	0.149	0.994	0.464	0.681	0.495	0.950
2	12-6-1	0.048	0.220	0.172	0.995	**0.317**	**0.563**	**0.366**	**0.966**
3	12-12-1	0.052	0.229	0.167	0.994	0.406	0.637	0.474	0.951
4	12-15-1	0.043	0.208	0.154	0.992	0.322	0.567	0.465	0.965
5	12-20-1	0.039	0.197	0.146	0.996	0.747	0.864	0.652	0.920

4. 2. LSTM Model

The THYAO stock price was predicted through a different structural LSTM model, and its result was compared. The LSTM model hyperparameters are shown in Table 6. In any of the layers, no dropout was used.

Table 6. LSTM and GRU model hyperparameters

Parameter	*Value*
Epoch	100
Batch size	30
Learning algorithm	adam
Delta	0.001
Loss function	MSE
Hidden layer transfer function	ReLU
Output layer transfer function	ReLU

The system learning process was carried out by creating different architectures for different LSTM units and output (ANN) units, and the results were transferred to Table 7. As can be seen in the table, the training and test performances of all models are very successful and close to each other. If evaluated according to the test performance, the best model 1 and 5 architectures are more successful than the others with 99% R-square value. Figure 13 shows the LSTM architecture training and validation loss graph. According to the graph, it is seen that the training error and validation values decrease together with the value of errors as the iteration progresses. The fact that the loss and validity error values decrease together and are very close to zero can be considered an indication of a successful and normal training process. Figure 13 seen that there is no overfitting or underfitting. In Figure 14, the test data and the actual values, and the predicted values are plotted together. As can be seen in the figure, the actual value and the predicted value are very close to each other and almost overlap. This show that system performance is good. This study also used a bidirectional LSTM model.

The bidirectional LSTM is a kind of LSTM. It consists of two LSTMs that perform bidirectional computation. One LSTM computes the array in the forward direction, and the other in the reverse direction. This boosts the efficiency of sequential problems such as time series, and text processing.

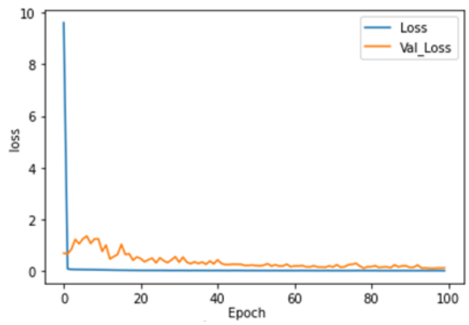

Figure 13. LSTM training error change graph

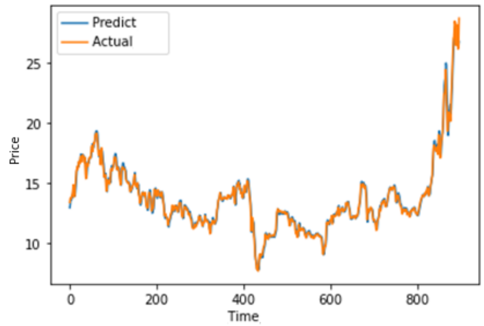

Figure 14. LSTM test comparison of actual and predicted values

Table 7. LSTM model machine learning network architecture, optimization parameters, training, and test performance

#	Network Architecture	Training Performance				Test Performance			
	LSTM-LSTM-Dense-Dense	MSE	RMSE	MAE	R^2	MSE	RMSE	MAE	R^2
1	64 - - - 1	0.014	0.120	0.084	0.998	0.143	0.378	0.258	0.985
2	64 - 64 - - 1	0.013	0.116	0.082	0.998	0.152	0.390	0.260	0.984
3	64 - - 32 - 1	0.013	0.113	0.080	0.998	0.199	0.446	0.291	0.979
4	64 - 64 - 32 - 1	0.013	0.115	0.080	0.998	0.159	0.399	0.260	0.983
5	64 unit BiLSTM- Dense(1)	0.012	0.109	0.076	0.998	0.130	0.361	0.246	0.986

4.3. GRU Model

System training and tests were performed using the parameters used in the LSTM for the GRU model, and the results were transferred to Table 8. As seen in the table, the R-square performances of all models are very close to each other (0.981 to 0.987). The best performance was the 2nd model with a value of 0.987. The change graph of the training loss and validity values of the system with repetition is given in Figure 15. In Figure 16, the test prediction values obtained from the trained system and the expected actual values are drawn together, and it is seen that they almost overlap with each other. This figure supports the success of training and testing numerical performance values.

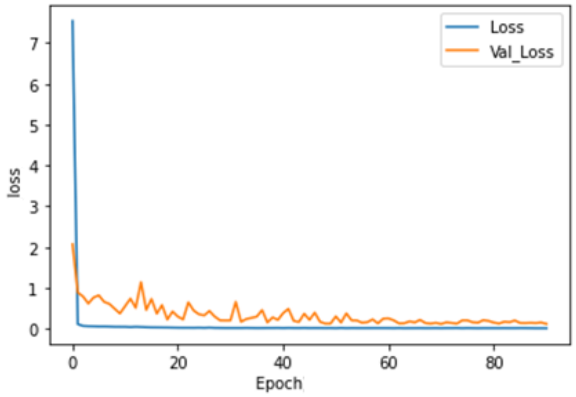

Figure 15. GRU model training error and validation error graph depended on the iteration

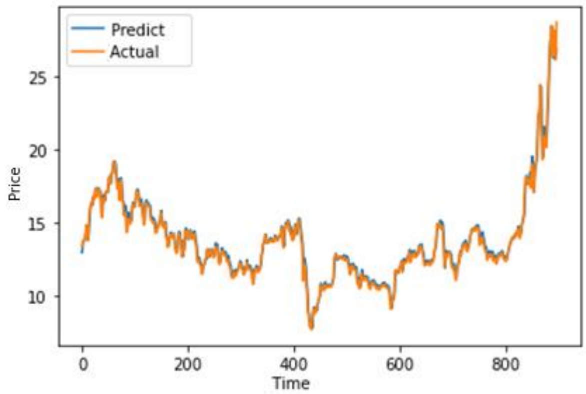

Figure 16. GRU model tests prediction values and actual values in displaying in the same figure

5. CONCLUSION

In this study, by using THYAO's 3036-day BIST data between the years 2010-2022, the prediction process was carried out by using different types of artificial neural networks, FNN, LSTM, and GRU machine learning methods. The data set was taken from tr.investing.com to *csv* format and preprocessed and turned into a supervised machine learning data set in the form of input and label. Used stock index values in the data set, the arithmetic average of the opening and closing index. Also, transaction volume values were added to the data set as an attribute to minimize possible speculative effects. To take into account the seasonal effects, seasonal information was added to the data set as an attribute. The first 70% of the data set was split for training and the remaining 30% for testing. The training and test datasets were transformed into supervised machine-learning datasets. During the transformation used 10 window sizes, so the data set has 10 attributes and a label. No scaling was done in the data set. Hence, the training and test MSE error amounts are in the range of 0.317 to 0.464. These values may be smaller when the data is scaled to a certain range. This does not affect the performance of the system. Considering R-Square and the test performance scale, it was seen that there was no significant difference between the FNN, LSTM, and GRU architectures. However, LSTM and GRU performed better, with 99% R-square. The change in LSTM and GRU architectures showed no significant change in the R-square performance value. But a performance difference of 92% to 97% was observed in the R-squared scale of the FNN architecture change. Hence, it can be concluded that the network architecture affects the system performance in FNN. When looking at the data set graph in Figure 7, it is seen that there is a big fluctuation, especially in the last data. Although this fluctuation is test data, the test R-square performance of the system has a very high value of 99%. This means that the system has been well-trained. The taught machine (software) can generate a forecast for the next days by using the data from the previous days. This will provide significant support to the investor while investing in a decision. However, many predictable/unpredictable factors affect the stock market index. While creating the data set, different factors that will affect the index can also be considered.

Table 8. GRU model machine architecture, optimization parameters, training, and test performance

#	Network Architecture	Training Performance				Test Performance			
	GRU- GRU- Dense- Dense	MSE	RMSE	MAE	R^2	MSE	RMSE	MAE	R^2
1	64 - - - 1	0.011	0.105	0.073	0.997	0.133	0.365	0.228	0.986
2	64 - 64 - - 1	0.011	0.106	0.073	0.998	0.126	0.355	0.236	0.987
3	64 - - 32 - 1	0.010	0.101	0.074	0.998	0.178	0.422	0.282	0.981
4	64 - 64 - 32 - 1	0.010	0.101	0.070	0.998	0.129	0.359	0.237	0.986
5	64 unit BiGRU-Dense(1)	0.011	0.104	0.072	0.995	0.174	0.417	0.295	0.981

In this study showed that machine learning can be an effective method for making forward projections based on the past data of the air transport sector. It can be concluded that LSTM and its derivative methods are more successful than the classical artificial neural network in the prediction index prediction of airport transportation sectors. ANN-based machine learning methods can be used successfully in the prediction of air transport and can provide long-term prescience to investors. In time series the structure of machine-learning methods and the effects of hyperparameters on the prediction performance were shown. More predictable forecasts can be made by taking into account unforeseen environmental factors such as economic policies, international relations, and interest rates.

REFERENCES

[1] Internet: M. Johnston, Biggest Companies in the World by Market Cap, https://www.investopedia.com/ biggest-companies-in-the-world-by-market-cap-5212784, 31.01.2022.

[2] P.N. Tan, M. Steinbach, V. Kumar, **Introduction to Data Mining**, Pearson Education, Inc, 2006, ISBN 0-321-42052-2, 2006.

[3] Ö. Duru, **Zaman Serileri Analizinde Arıma Modelleri ve Bir Uygulama**, Yüksek Lisans Tezi, İstanbul Üniversitesi, Sosyal Bilimler Enstitüsü, 2007.

[4] S. Hansun, "A new approach of moving average method in time series analysis", **Conference on New Media Studies (CoNMedia)**, Tangerang, Indonesia, 2014.

[5] R. J. Hyndman and G. Athanasopoulos, **Forecasting: Principles and Practice**. OTexts Publisher, 2018.

[6] J. F. Torres, D. Hadjout, A. Sebaa, F. Martínez-Álvarez, and A. Troncoso, "Deep Learning for Time Series Forecasting: A Survey", *Big Data*, 9(1), 2011.

[7] O. Kaynar and S. Taştan, "Comparasion of MLP Artifical Neural Network and Arima Method in Time Series Analysis", *Erciyes Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi*, 33, 161-172, 2009.

[8] B. K. Bayraktar, and B. Badur B. "Yapay Sinir Ağları ile Borsa Endeksi Tahmini", *Yönetim*, 20(63), 2009.

[9] H. Aygören, H. Sarıtaş, T. Moralı, "İMKB 100 Endeksinin Yapay Sinir Ağları ve Newton Nümerik Arama Modelleri ile Tahmini", *International Journal of Alanya Faculty of Business*, 4(1), 73-88, 2012.

[10] S. Siami-Namini, N. Tavakoli, A. S. Namin, "A Comparison of ARIMA and LSTM in Forecasting Time Series", **17th IEEE International Conference on Machine Learning and Applications**, IEEE, 1394-1401, 2018.

[11] M. Yücesan, "Sales Forecast with YSA, ARIMA and ARIMAX Methods: An Application in the White Goods Sector", *Journal of Business Research-Türk*, DOI: 10.20491/isarder.2018.414, 2018.

[12] S. Kalyoncu, A. Jamil, E. Karatas, J. Rasheed, C. Djeddi, "Stock Market Value Prediction using Deep Learning", 3**rd International Conference on Data Science and Applications (ICONDATA'20)**, 2020.

[13] E. Çınaroğlu, T. Avcı, "THY Hisse Senedi Değerinin Yapay Sinir Ağları İle Kestirimi", *Atatürk Üniversitesi İktisadi ve İdari Bilimler Dergisi*, 34(1), 1-20, 2020.

[14] D. Güleryüz, E. Özden, Ü. Gülhan, "Predicting BIST 30 Index with ARIMA and RNN-LSTM Models", **24. Finans Sempozyumu**, 2021.

[15] S. Ranjan, P. Kayal and M. Saraf, "Bitcoin Price Prediction: A Machine Learning Sample Dimension Approach", *Computational Economics*, DOI: 10.1007/s10614-022-10262-6, 2022.

[16] R. Solgi, H. A. Lo´ aiciga and M. Kram, "Long short-term memory neural network (LSTM-NN) for aquifer level time series forecasting using in-situ piezometric observations", *Journal of Hydrology*, 601, 2021.

[17] B. Lindemann, T. Müller, H. Vietz, N. Jazdi, M. Weyrich, "A survey on long short-term memory networks for time series prediction", *Pocedia CIRP*, (99), 650-655, 2021.

[18] U. Demirel, H. Cam, R. Unlu, "Predicting Stock Prices Using Machine Learning Methods and Deep Learning Algorithms: The Sample of the Istanbul Stock Exchange", *Gazi University Journal of Science*, 34 (1): 63-82, 2021.

[19] Internet: THYAO, Türk Hava Yolları Anonim Ortaklığı, https://tr.investing.com/equities/turk-hava-yollari-historical-data, 02.05.2022.

[20] J. Zou, Y. Han, S. S. So, Overview of Artificial Neural Networks. In: Livingstone D.J. (Eds.) Artificial Neural Networks. Methods in Molecular Biology™, 458, 14-22, Humana Press, 2008.

[21] Z. R. Yang, and Z. Yang, **Artificial Neural Networks. Comprehensive Biomedical Physics**, Elsevier. Editor(s): Anders Brahme, 1-17, 2014.

[22] K. Zarzycki, and M. Lawrynczuk, "LSTM and GRU Neural Networks as Models of Dynamical Processes Used in Predictive Control: A Comparison of Models Developed for Two Chemical Reactors", *Sensors*, 21(16), 2021.

[23] M. B. Er, İ. Işık, "LSTM Tabanlı Derin Ağlar Kullanılarak Diyabet Hastalığı Tahmini", *Türk Doğa ve Fen Dergisi*, 10(1), 68-74, 2021.

[24] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory", *Neural Computation*, 9 (8): 1735–1780, 1997.

[25] X. H. Le, H. V. Ho, G. Lee, S. Jung, "Application of Long Short-Term Memory (LSTM) Neural Network for Flood Forecasting", *Water*, 11, 1387, 2019.

[26] S. Tanışman, A. A. Karcıoğlu, A. Uğur, H. Bulut, "LSTM Sinir Ağı ve ARIMA Zaman Serisi Modelleri Kullanılarak Bitcoin Fiyatının Tahminlenmesi ve Yöntemlerin Karşılaştırılması", *European Journal of Science and Technology*, 32, 514-520, 2021.

[27] K. Cho, B. Merrienboer, D. Bahdanau, Y. Bengio, On the Properties of Neural Machine Translation: Encoder-Decoder Approaches, *arXiv:1409.1259*, 2014.

[28] K. Cho, B. Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, Y. Bengio, "Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation", *arXiv:1406.107*, 8 (3) , 2014.