STATISTICAL TRAINING OF LONG SHORT-TERM MEMORY ARTIFICIAL NEURAL NETWORKS BY USING PARTICLE SWARM OPTIMIZATION FOR FORECASTING

Eren Bas – Erol Egrioglu

Abstract

Long short-term memory artificial neural networks are a type of artificial neural network that is frequently used in forecasting problems. Like many artificial neural networks, one of the successful performances of Long short-term memory artificial neural networks is the training process of the network. Many training algorithms of that network are based on the nonlinear least square and the distribution of error terms is not used in these algorithms. In this study, maximum likelihood estimators for long short-term memory artificial neural networks are obtained with Gumbel distribution based on particle swarm optimization. The analysis performance of the proposed method is evaluated over the closing prices of the German and Spanish stock exchanges. The performance evaluation of the proposed method is compared with both many deep and shallow neural models. According to the results of the evaluation, the proposed statistical-based Long short-term memory artificial neural network has a superior forecasting performance than the other compared methods.

Key words: long short-term memory, particle swarm optimization, statistical training, forecasting.

JEL Code: C45, C53

Introduction

The analysis methods used for the time series forecasting problem can be classified as classical forecasting methods, shallow neural networks, and deep artificial neural networks. While classical time series analysis methods are mainly used in the analysis of linear time series, shallow and deep artificial neural networks are effectively used in the analysis of nonlinear time series. Shallow neural networks have generally only one or two layers between the input and output layers. However, deep neural networks have more than one hidden layer between the input and hidden layer. It can be also said that the number of parameters to be optimized in the training of shallow neural networks is less than the number of parameters to be optimized in the training of deep artificial neural networks. An important advantage of deep neural networks over shallow neural networks is the use of parameter sharing. By using parameter sharing, the training time of the network is shortened and fewer parameters are optimized during the training process of the network. The most important advantage of parameter sharing is to determine what different time steps mean. When parameter sharing is used, each time step only needs to be learned once. One of the other important differences between deep neural networks from shallow neural networks is that they use previous time steps of the time series as input. These time steps are obtained by shifting the target values by one time step. Deep artificial neural networks are distinguished from shallow artificial neural networks by these features.

Long short term artificial neural networks (LSTM-ANN), one of the most popular deep artificial neural networks, are frequently used in the literature for time series forecasting problems. The LSTM-ANN proposed by Hochreiter and Schmidhuber (1997) uses a gating mechanism to solve the vanishing or exploding gradient problem in the backpropagation learning algorithm used in the training of recurrent neural networks. One reason for this is that the back-propagation learning algorithm has difficulty in learning long-term dependency structures.

An important problem of LSTM-ANN is to determine the optimization algorithm used in the training of the network. Although the Adam algorithm is a frequently used optimization algorithm in the training of the network, in recent years, many different artificial intelligence optimization algorithms have been used in the training of the network. Chung and Shin (2018) used a genetic algorithm in the training of LSTM-ANN for stock market forecasting. Peng et al. (2018) used a differential evolution algorithm in the training of LSTM-ANN for electricity price forecasting. Shao et al. (2019) used particle swarm optimization (PSO) in the training of LSTM-ANN for Nickel price forecasting. Yang and Wei (2020) used improved chaotic lion swarm optimization in the training of LSTM-ANN for forecasting problem. Yang et al. (2020) used an improved whale optimization algorithm in the training of LSTM-ANN for carbon price forecasting. Shahid et al. (2021) used a novel genetic algorithm in the training of LSTM-ANN for forecasting of wind power. Tuerxun et al. (2022) used a modified bald eagle search algorithm in the training of LSTM-ANN for wind power forecasting. Ulum and Girsang (2022) used a symbiotic organism search algorithm in the training of LSTM-ANN for stock forecasting. All these training algorithms used in the training of LSTM-ANN are based on nonlinear least squares. In the use of all these training algorithms, the distribution of error terms in Long short-term memory models, as in many artificial neural network models, is ignored.

In this study, a statistical learning algorithm is proposed for the first time for long short-term memory artificial neural networks. Maximum likelihood estimators for long short-term memory artificial neural networks are obtained with Gumbel distribution. The particle swarm optimization algorithm proposed by Kennedy and Eberhart (1995) is used in maximizing the likelihood function when the error terms have Gumbel distribution. The forecasting performance of the proposed new method is evaluated over the closing values of German and Spain stock exchanges and the proposed method is compared with various shallow and deep artificial neural networks.

1 The Proposed Method

LSTM-ANN is a type of recurrent neural network that can be used in image processing, clustering, and forecasting problems. Information in the LSTM-ANN can be stored through gates in the network structure. The outputs of the input gate, forget gate, cell candidate, and the cell state are calculated by Equations (1) and (4).

$$\mathrm{i_t} = \sigma(\mathrm{W_i x_t} + \mathrm{R_i h_{t-1}} + \mathrm{b_i}) \qquad (1)$$

$$\mathrm{f_t} = \sigma(\mathrm{W_f x_t} + \mathrm{R_f h_{t-1}} + \mathrm{b_f}) \qquad (2)$$

$$\mathrm{g_t} = \sigma\left(\mathrm{W_g x_t} + \mathrm{R_g h_{t-1}} + \mathrm{b_g}\right) \qquad (3)$$

$$c_t = f_t * c_{t-1} + i_t * g_t \qquad (4)$$

Besides, the output of the output gate and the hidden state value is calculated by Equations (5) and (6).

$$o_t = \sigma(W_o x_t + R_o h_{t-1} + b_o) \qquad (5)$$

$$h_t = o_t * \tanh(c_t) \qquad (6)$$

And the final output of the network is calculated by Equation (7).

$$\hat{y}_t = \sigma(W_{FC} h_t + b_{FC}) \qquad (7)$$

In this study, for the first time in the literature, LSTM-ANN is written with random error terms. Thus, by adding an error term to the LSTM-ANN, the model is turned into a statistical model Besides, the likelihood function is defined for LSTM-ANN with Gumbel distribution.

Moreover, the PSO algorithm is used while maximizing that likelihood function. The algorithm of the proposed method is given as follows.

Algorithm 1. The proposed method

Step 1. Determining the parameters of the proposed method

m: the number of time steps

n: The number of hidden layers

p: the number of inputs

$maxitr$: Maximum iteration number

pn: The number of particles

Step 2. The generation of initial velocity and position values in PSO

A particle in PSO consists of $4m(p+m+1)+m+2$ positions. This total number of positions is calculated based on the time step and the number of inputs. For the proposed method, $4m(p+m+1)+m+1$ of these positions are about LSTM-ANN parameters and the last position is for the standard deviation (β) of the Gumbel distribution. The positions consist of $W_i: p \times m$, $R_i: m \times m$, $b_{\mathrm{i}}: 1xm$, $W_f: p \times m$, $R_f: m \times m$, $b_{\mathrm{f}}: 1xm$, $W_g: p \times m$, $R_g: m \times m$, $b_{\mathrm{g}}: 1xm$, $W_o: p \times m$, $R_o: m \times m$, $b_{\mathrm{o}}: 1xm$, $W_{FC}: m \times 1$ and $b_{FC}: 1 \times 1$ and β: 1×1. All these positions are generated with the help of a uniform distribution with [0,1] parameters. The positions and velocities of a particle can be described as $P_{i,j}^{(t)}$ and $V_{i,j}^{(t)}$ respectively. Here, $P_{i,j}^{(t)}$ and $V_{i,j}^{(t)}$ are the j^{th} position and velocity of the i^{th} particle at the t^{th} iteration.

Step 3. Calculate the fitness function value of each position.

While the distribution of error terms ε_t, are Gumbel and $\varepsilon \sim Gumbel(0, \beta I)$, the probability density function of an error term can be written as in Equation (8) and the likelihood function is given in Equation (9).

$$f(\varepsilon_t) = \frac{1}{\beta} exp\left(-\frac{\varepsilon_t}{\beta} - exp\left(-\frac{\varepsilon_t}{\beta}\right)\right) \qquad (8)$$

$$L = \frac{1}{\beta^n} exp\left(-\sum_{t=1}^{n}\left(\frac{\varepsilon_t}{\beta} + exp\left(-\frac{\varepsilon_t}{\beta}\right)\right)\right) \qquad (9)$$

The fitness function is calculated by Equation (10).

$$\mathrm{L} = \frac{1}{\beta^{n}} \exp\left(-\sum_{t=1}^{n}\left(\frac{Y_t-\sigma(W_{FC}h_t+b_{FC})}{\beta} + \exp\left(-\frac{Y_t-\sigma(W_{FC}h_t+b_{FC})}{\beta}\right)\right)\right) \quad (10)$$

Step 4. Obtain Pbest and gbest.

gbest and Pbest is are the best state of all particles and the best positions of particles respectively and they determined according to the fitness function value given in Equation 10.

Step 5. Calculate the cognitive, social coefficients, and inertia weight parameters of PSO as given in Equations (11) - (13) respectively.

$$c_1^{(t)} = (c_1^{final} - c_1^{initial})\frac{t}{maxitr} + c_1^{initial} \quad (11)$$

$$c_2^{(t)} = (c_2^{initial} - c_2^{final})\frac{maxitr-t}{maxitr} + c_2^{final} \quad (12)$$

$$w^{(t)} = (w^{initial} - w^{final})\frac{maxitr-t}{maxitr} + w^{final} \quad (13)$$

Step 6. Calculate the new velocities and positions as given in Equations (14) and (15) respectively.

$$V_{i,j}^{(t)} = w^{(t)}V_{i,j}^{(t-1)} + c_1^{(t)}r_1\left(Pbest_{i,j}^{(t)} - P_{i,j}^{(t)}\right) + c_2^{(t)}r_2\left(gbest_j^{(t)} - P_{i,j}^{(t)}\right) \quad (14)$$

$$P_{i,j}^{(t)} = P_{i,j}^{(t-1)} + V_{i,j}^{(t)} \quad (15)$$

Step 7. Update Pbest and gbest by considering Equation (10).

Step 8. If the maximum number of iterations is reached the algorithm is finished.

2 Applications

The applications of the proposed method were carried out on the closing values of Spain (IBEX) and German stock markets (DAX) in 2015. The graphs of each time series are given in Figs 1 and 2. Each data set is separated as training, validation and test set. The validation and test set lengths of each method are taken as twenty-five. The analysis performance of the proposed method is evaluated with LSTM-ANN, gated recurrent unit artificial neural networks (GRU-

ANN), Pi-Sigma artificial neural networks (PS-ANN) and single multiplicative neuron model artificial neural networks (SMNM-ANN).

Fig. 1: Development of IBEX time series

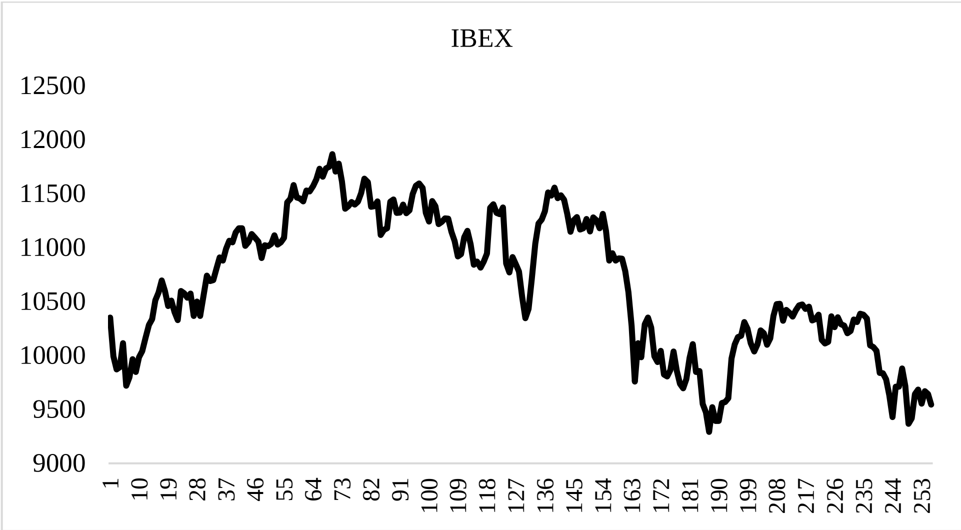
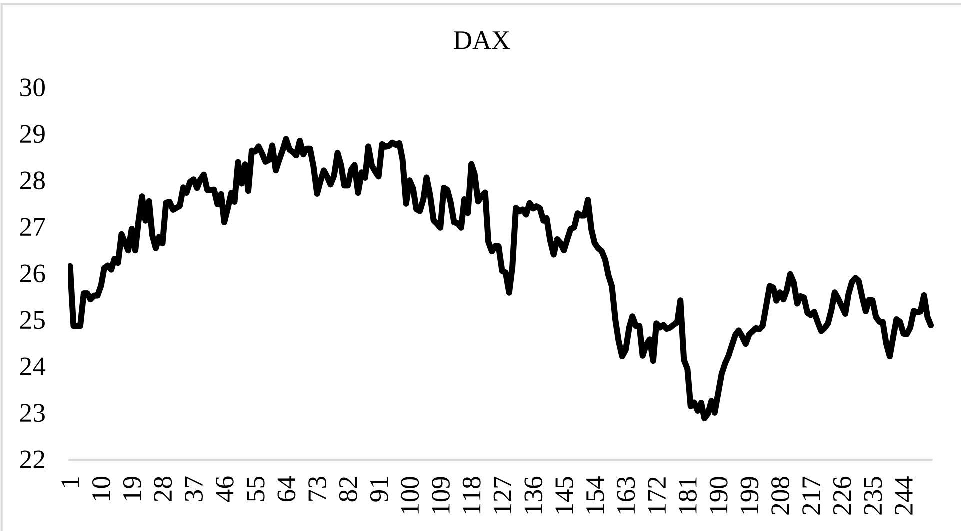


Fig. 2: Development of DAX time series



In the application of these methods, the number of inputs is changed between one and five with an increment of one, the number of hidden layer units is changed between one and five with an increment of one, the number of hidden layers is changed between one and two with an increment of one. Each method is applied ten times using random weights and mean, standard deviation (Std. Dev.), interquartile range (IQR), minimum and maximum statistics are obtained by considering the root mean square error (RMSE) criterion. The results of all analyses are given in Tab. 1.

Tab. 1: The RMSE statistics for all methods

Data Sets	Methods	Mean	Median	Std. Dev.	IQR.	Minimum	Maximum
IBEX	LSTM-ANN	537,3942	533,0776	26,3255	34,3892	492,6494	610,2244
	GRU-ANN	226,8032	217,9296	32,4058	26,9833	192,3411	309,4504
	PS-ANN	171,4421	165,3098	19,5364	25,5221	145,4245	227,1981
	SMNM-ANN	209,1294	218,9701	40,2669	67,7748	155,5859	261,6200
	Proposed Method	146,4961	146,6099	1,5401	0,8303	142,9996	148,9930
DAX	LSTM-ANN	0,4592	0,4600	0,0354	0,0569	0,4085	0,5508
	GRU-ANN	0,4311	0,4143	0,0411	0,0595	0,3749	0,4984
	PS-ANN	0,3417	0,3159	0,0764	0,0477	0,2778	0,5272
	SMNM-ANN	0,3774	0,4118	0,0682	0,1351	0,2784	0,4368
	Proposed Method	0,2765	0,2749	0,0035	0,0022	0,2745	0,2857

When the analysis results are examined, it is seen that the proposed method is the most successful method for the analysis of both the DAX and IBEX time series. In addition, the proposed method is the best method according to all statistics.

Conclusion

Although LSTM-ANN produces successful results in forecasting problems in the literature, as in many artificial neural network models, the error term has been neglected in the model of the network. In this study, for the first time in the literature, LSTM-ANN is transformed into a statistical model by adding an error term based on the Gumbel distribution. The likelihood function is obtained under the Gumbel distribution for the LSTM-ANN model. The proposed new LSTM-ANN method can be used to analyse non-seasonal time series. Besides, the proposed algorithm uses PSO as an optimization technique in the training process. According to the results of the analysis, it is seen that the proposed LSTM-ANN model, in which the error term is included in the model, produces better results than the other methods compared. In future studies, different error terms can be included in the LSTM-ANN model. Different artificial intelligence optimisation algorithms can be used in the training of the proposed new method.

References

Chung, H., & Shin, K. S. (2018). Genetic algorithm-optimized long short-term memory network for stock market prediction. *Sustainability*, *10*(10), 3765.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, *9*(8), 1735-1780.

Kennedy, J., & Eberhart, R. (1995, November). Particle swarm optimization. In *Proceedings of ICNN'95-international conference on neural networks* (Vol. 4, pp. 1942-1948). IEEE.

Peng, L., Liu, S., Liu, R., & Wang, L. (2018). Effective long short-term memory with differential evolution algorithm for electricity price prediction. *Energy*, *162*, 1301-1314.

Shahid, F., Zameer, A., & Muneeb, M. (2021). A novel genetic LSTM model for wind power forecast. *Energy*, *223*, 120069.

Shao, B., Li, M., Zhao, Y., & Bian, G. (2019). Nickel price forecast based on the LSTM neural network optimized by the improved PSO algorithm. *Mathematical Problems in Engineerin*g, 1934796.

Tuerxun, W., Xu, C., Guo, H., Guo, L., Zeng, N., & Gao, Y. (2022). A wind power forecasting model using LSTM optimized by the modified bald eagle search algorithm. *Energies*, *15*(6), 2031.

Ulum, D. S. N., & Girsang, A. S. (2022). Hyperparameter optimization of long-short term memory using symbiotic organism search for stock prediction. *International Journal of Innovative Research and Scientific Studies*, *5*(2), 121-133.

Yang, S., Chen, D., Li, S., & Wang, W. (2020). Carbon price forecasting based on modified ensemble empirical mode decomposition and long short-term memory optimized by improved whale optimization algorithm. *Science of the Total Environment*, *716*, 137117.

Yang, Z., & Wei, C. (2020). Prediction of equipment performance index based on improved chaotic lion swarm optimization–LSTM. *Soft Computing*, *24*, 9441-9465.

Contact

Eren Bas
Giresun University
Faculty of Arts and Science, Department of Statistics, 28200, Giresun, Türkiye
eren.bas@giresun.edu.tr

Erol Egrioglu
Giresun University
Faculty of Arts and Science, Department of Statistics, 28200, Giresun, Türkiye
erol.egrioglu@giresun.edu.tr