NESTED AND MIXED ARTIFICIAL BEE COLONY ALGORITHM FOR LONG SHORT TERM MEMORY DEEP ARTIFICIAL NEURAL NETWORK

Erol Egrioglu – Eren Bas

Abstract

Deep neural networks have become an important alternative to the forecasting problem. Deep neural networks have the ability to adapt better to complex and nonlinear surfaces than shallow neural networks. One of the most important problems of deep neural networks is the process of determining the optimal hyperparameters. This process is carried out outside the training algorithm and is usually a process that is operated by trial and error or subjective decisions. In this study, nested and mixed artificial bee colony algorithm is proposed for the first time in the literature and a forecasting algorithm that tries to reach the optimal solution by reducing the hyperparameter determination and training process to a single process for long short term memory, one of the important deep artificial neural networks. The proposed algorithm is applied to real life time series in terms of computation time and prediction accuracy and the results are discussed.

Key words: Forecasting, Long Short Term Memory, Deep Learning, Artificial bee colony

JEL Code: C45, C53

Introduction

Deep feedback artificial neural networks offer a good alternative to solve the prediction problem. The feedback mechanism gives the neural network a dynamic structure, allowing the modelling to reflect the change of learning patterns over time. The deep neural network with the simplest structure in the literature is the simple recurrent neural network (SRNN), a dynamic version of the multilayer perceptron. The problem of exploding and resetting gradients in deep feedback networks has led to alternative solutions. Long short-term memory deep neural network (LSTM) (Hocreiter and J. Schmidhuber, 1997) is an artificial neural network that was created as a solution to these problems. LSTM is an artificial neural network that was introduced as a solution to these problems. In addition to LSTM, different deep artificial neural networks have also been used to solve the prediction problem. gated recurrent unit deep neural networks (GRU) (Cho et al., 2014) and convolutional neural networks (CNN) (Kunihiko F.,1980) are two important artificial neural network models used in solving the prediction problem. CNN has a feed-forward architecture and is not a feedback network. GRU is a simplified version of LSTM. Egrioglu and Bas (2024) introduced another feedback deep artificial neural network (DeepDenT) to the literature. DeepDenT neural network has a dendritic cell structure and is able to improve the prediction performance by multiplicative combinations of inputs.

In deep recurrent neural networks, in addition to the input or feature selection problem, the number of time steps and the number of hidden layers must also be determined. In deep neural networks, which require more parameters than shallow neural networks, the selection of hyperparameters is a bigger problem. In addition, the training of deep neural networks is another topic of interest in the literature. The most widely used training algorithms in the literature are gradient-based algorithms. In these algorithms, the objective function must be differentiable at certain points along the iterations. In addition, these gradient-based algorithms have a high probability of getting stuck in local optimum traps. In addition to all these, gradient-based problems arise in deep neural networks. In the literature, the most well-known gradient-based algorithms for training deep neural networks are AdaGrad algorithm by Duchi et al. (2014) and Adam algorithm by Kingma and Ba (2014). In the training of deep feedback networks, artificial intelligence optimization algorithms have become preferable training algorithms because they do not need a gradient. Bas et al. (2022) proposed a training algorithm based on particle swarm optimization for training a simple feedback neural network. Cansu et al. (2023) proposed an algorithm based on particle swarm optimization for training the LSTM artificial neural network. One of the deep feedback artificial neural networks is the artificial bee colony optimization algorithm proposed by Karaboga (2005). The artificial bee colony algorithm is known to produce successful results in the training of artificial neural networks. In this study, for the first time in the literature, the nested and mixed artificial bee colony algorithm is proposed as an algorithm that can perform the training and hyperparameter selection of the LSTM artificial neural network together. In the second section of the study, the LSTM artificial neural network is summarized, and in the third section, the nested and mixed artificial bee colony algorithm proposed for LSTM is introduced.

1 Long short Term Memory Artificial Neural Network

The LSTM neural network has a hierarchical architecture consisting of a combination of LSTM cells. The output of the LSTM cell is calculated by formulas (1)-(6).

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

$$f_t = \sigma\left(W_f x_t + R_f h_{t-1} + b_f\right) \tag{2}$$

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

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

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

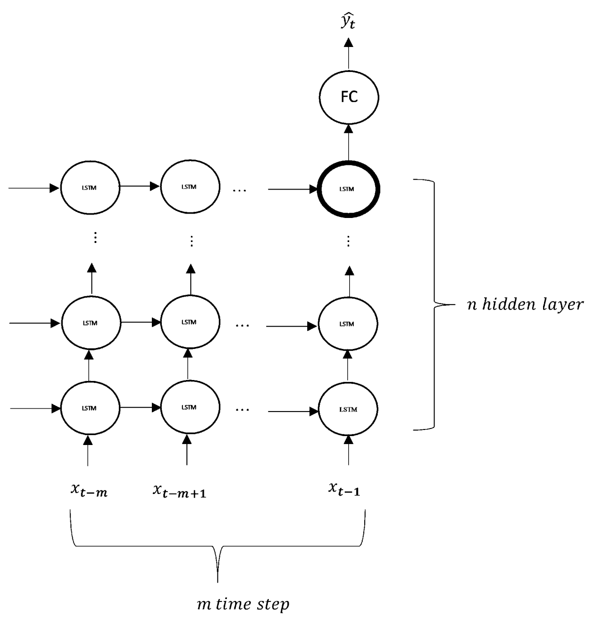
$$h_t = o_t * \tanh(c_t) \tag{6}$$

The architecture of the LSTM neural network is given in Figure 1. This architecture is specifically designed for the forecasting problem and was proposed in Cansu et al. (2023). The input of a LSTM cell is lagged variable of $x_t = (y_t, y_{t-1,\dots,}y_{t-p+1})$.

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

The output of the artificial neural network shown in Figure 1 is calculated by the formula (7) based on the output of the neuron given in bold lines in the figure.

Fig. 1: The LSTM Architecture



2 Nested and Mixed Artificial Bee colony for LSTM

It is possible to perform the training as well as the hyperparameter optimization of the LSTM neural network simultaneously within the same optimization algorithm. In this case, by choosing the trial and error method for hyperparameter selection, it is possible to complete the process in a shorter time without both the problem of working with a solution far from the optimal solution and trying all possible results. Therefore, nested and mixed artificial bee colony algorithm is proposed in this study. The steps of the nested and mixed artificial bee colony algorithm are summarized below.

Algorithm. Nested and Mixed Artificial Bee Colony Algorithm for Training of LSTM

Step 1. The boundaries of the hyperparameter space are determined.
pg_{max} is the maximum number of lagged variables number ($pg \in [1, pg_{max}]$)
m_{max} is the maximum number of hidden layer number ($m \in [1, m_{max}]$)
h_{max} is the maximum number of time steps number ($h \in [1, h_{max}]$)

Step 2. The source positions in the outer loop are randomly generated. These positions correspond to hyperparameters.

$$X_{i1}^{outer} \sim Uniform(pg_{max}) \tag{8}$$
$$X_{i2}^{outer} \sim Uniform(m_{max}) \tag{9}$$
$$X_{i3}^{outer} \sim Uniform(h_{max}) \tag{10}$$

Step 3. The fitness value is calculated in the hyperparameter setup corresponding to each source in the outer loop. For this, the operations in the inner loop for the LSTM architecture created with hyper-parameters corresponding to the source positions in the inner loop are applied in Steps 3.1-3.7.

Step 3.1. The source positions in the inner loop are randomly generated. These positions correspond to weights of the LSTM in equations (1), (2), (3), (5) and (7).

Step 3.2. The outputs of the LSTM are calculated by using (1)-(7) and training data set for each sources and the mean square errors (MSE) criterion are calculated for each sources.

Step 3.3. The best sorces is determined and it is saved as $bestsw$.

Step 3.4. The worker bee phase is implemented in the inner loop.

Step 3.5. The scout bee phase is applied in the inner loop.

Step 3.6. The best sources are updated. The stopping rule is checked.

Step 3.7. The explorer bee phase is applied in the inner loop.

Step 4. The best source is determined and it is saved as $bests$.

Step 5. The worker bee phase is implemented in the outer loop.

Step 6. The scout bee phase is applied in the outer loop.

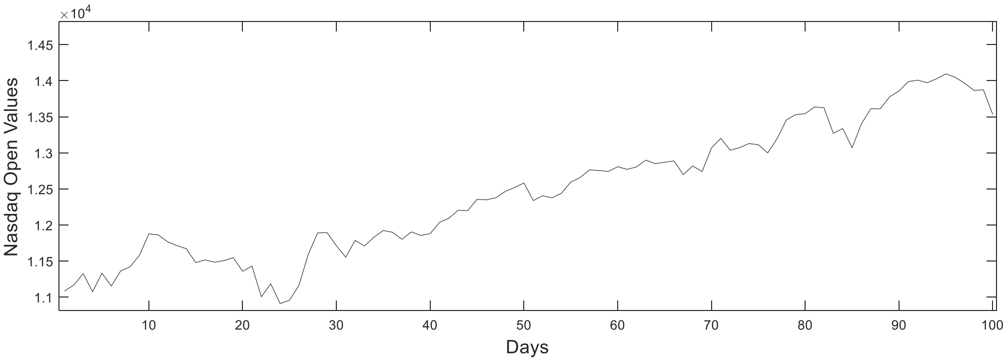
Step 7. The best sources are updated. The stopping rule is checked.

Step 8. The explorer bee phase is applied in the outer loop.

3 Applications

In the application, the 100-day Nasdaq stock market open interest time series between September 29, 2020 and February 22, 2021 is used. The graph of the time series is given in the Figure 2.

Fig. 2: The Nasdaq Opening Values



In order to compare the performance of the proposed method, a multilayer perceptron (MLP) as a classical artificial neural network, SRNN, GRU and LSTM as deep artificial neural networks were used. In the implementation of these alternatives, hyperparameter selection was performed by trial and error method.

The results obtained from the application are given in table 1. The ANN architecture obtained for the best hyperparameters of all methods was trained with 30 different random initial weights and the root of mean square errors (RMSE) values for the test set were calculated and the statistics of the RMSE measures are given in Table 1. In the application, 3 different test set lengths of 10, 20 and 30 were used.

Table 1. Results obtained in the application of the methods

ntest	Method	Mean	Median	Minimum	Maximum	pgbest	mbest	hbest
10	MLP	194.63	194.3138	187.90	204.77	5	-	5
	LSTM	194.63	194.3138	187.90	204.77	4	1	1
	SRNN	194.79	193.9318	191.73	205.08	3	-	1
	GRU	191.68	191.5730	189.28	196.42	1	2	2
	Proposed Method	139.06	139.02	137.96	140.97	3	1	1
20	MLP	222.65	222.59	220.04	227.07	4	-	4
	LSTM	222.65	222.59	220.04	227.07	1	1	1
	SRNN	222.89	222.31	220.97	228.32	4	-	1
	GRU	230.57	228.68	218.32	251.95	1	1	2
	Proposed Method	170.40	170.41	169.98	170.79	3	1	1
30	MLP	203.86	204.05	202.58	205.12	4	-	5
	LSTM	203.86	204.05	202.58	205.12	1	1	1
	SRNN	212.03	208.62	204.74	232.81	5	-	1
	GRU	205.36	203.94	200.88	212.23	2	1	3
	Proposed Method	157.02	157.02	156.91	157.22	1	1	1

When the RMSE statistics given in Table 1 are analyzed, it is seen that the proposed method produces better results than the other methods in all three test set lengths.

Conclusion

In this study, nested and mixed artificial bee colony is proposed as a new optimization algorithm. The proposed method is capable of simultaneously training the LSTM neural network and selecting hyperparameters. The performance of the proposed method is compared with MLP, LSTM, GRU and SRNN, which are the most commonly used ANN methods in the literature. It is observed that the performance of the proposed method in training the LSTM neural network is better than the particle swarm optimization presented in Cansu et al. (2023). In addition, a 70% improvement in calculation time was achieved compared to the trial and error method. In the case of using the proposed method, better performance results were obtained than other ANNs such as SRNN, MLP and GRU. In future studies, the proposed method will be applied on different time series to investigate the performance of the method.

Acknowledgment

This study is funded by the Turkish Scientific and Technological Research Council (TÜBİTAK) 2224-A - Grant Program for Participation in Scientific Meetings Abroad.

References

Bas, E., Egrioglu, E. & Kolemen, E. (2022) Training simple recurrent deep artificial neural network for forecasting using particle swarm optimization. *Granul. Comput.* **7**, 411–420.

Cansu, T., Kolemen, E., Karahasan, Ö., Bas, E., & Egrioglu, E. (2023). A new training algorithm for long short-term memory artificial neural network based on particle swarm optimization. *Granular Computing*, *8*(6), 1645-1658.

Duchi, J., Hazan, E., & Singer, Y. (2011). Adaptive subgradient methods for online learning and stochastic optimization. *Journal of machine learning research*, *12*(7).

Egrioglu, E., Bas, E. (2024). A new deep neural network for forecasting: Deep dendritic artificial neural network. Artif Intell Rev 57, 171.

Hocreiter, S., Schmidhuber, J. (1997). Long Short-Term Memory, *Neural Computation* 9(8) 1735–1780.

Cho, K., Van M. B., Bahdanau, D., Bougares, F., Schwenk H., Bengio Y. (2014). Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. Association for Computational Linguistics. arXiv:1406.1078.

Karaboga, D. (2005). An idea based on honey bee swarm for numerical optimization. TR-06, Erciyes University, Engineering Faculty, Computer Engineering Department.

Kingma, D., Ba, J.L. (2014). Adam: a method for stochastic optimization. Comput Sci 1–15. arXiv: 1412.6980.

Kunihiko F. (1980). Neocognitron: A Self-organizing Neural Network Model for a Mechanism of Pattern Recognition Unaffected by Shift in Position. Biol. Cybernetics 36, 193 202.

Contact

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

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