An intelligent demand forecasting model using a hybrid of metaheuristic optimization and deep learning algorithm for predicting concrete block production

Huthaifa Al-Khazraji[1], Ahmed R. Nasser[1], Sohaib Khlil[2]
[1]Department of Control and System Engineering, University of Technology-Iraq, Baghdad, Iraq
[2]Department of Applied Mechanics, Technical College-Baghdad, Middle Technical University, Baghdad, Iraq

Article Info

Article history:

Received Aug 14, 2021
Revised Feb 4, 2022
Accepted Feb 23, 2022

Keywords:

Deep learning
Demand forecasting
Firefly algorithm
Gated recurrent units
Metaheuristic optimization

ABSTRACT

Demand forecasting aims to optimize the production planning of industrial companies by ensuring that the production planning meets the future demand. Demand forecasting utilizes historical data as an input to predict future trends of the demand. In this paper, a new approach for developing an intelligent demand forecasting model using a hybrid of metaheuristic optimization and deep learning algorithm is presented. Firefly algorithm-based gated recurrent units (FA-GRU) is used to tackle the production forecasting problem. The proposed model has been evaluated and compared with the standard gated recurrent unit (GRU) and standard long short-term memory model (LSTM) using historical data of 36 months of concrete block manufacturing at dler company in Iraq. The prediction accuracy of the three models is evaluated using the root mean square error (RMSE), the mean absolute percentage error (MAPE) and the statistical coefficient of determination (R^2) indicators. The outcomes of the study show that the proposed FA-GRU gives better forecasting results compared to the standard GRU and standard LSTM.

Corresponding Author:

Huthaifa Al-Khazraji
Control and System Engineering Department, University of Technology-Iraq
Baghdad, Iraq
Email: 60141@uotechnology.edu.iq

1. INTRODUCTION

Demand forecasting is an important research area for production companies to ensure that the production planning meets the future demand. Due to demand uncertainty, production companies usually suffer from many problems such as excessive production and out-of-stock (stockouts) because of an over-anticipation of the demand and an under-anticipation of the demand respectively [1], [2]. Therefore, it is necessary to have accurate demand forecasting to have a sustainable competitive advantage in the industrial market. For this purpose, the historical data set of sales is used as an input to develop a prediction model to predict future trends of demand [3]. Many methods and techniques have been developed in the area of demand forecasting. In terms of traditional forecasting methods, autoregressive integrated moving average (ARIMA) is considered the most popular time series model due to its simplicity and flexibility. However, the ARIMA is limited to a linear time series data set. To overcome this problem, neural networks (NNs) including machine learning (ML) and deep learning (DL) based models have gained more attention and have effectively applied to nonlinear time series forecasting.

Bousqaoui *et al.* [1] presented a comparative study of four prediction models based on a real-life dataset taken from a supermarket in Morocco. The models are the ARIMA, the multi-layer perceptron

(MLP), the long short-term memory (LSTM) and the convolutional neural network (CNN). The finding was that the CNN model is more successful than the other three models. In the same way, Martínez Cervera *et al*. [4] compared four different forecasting models to estimate the different alternatives, trends and needs of the student population in Colombia. The models are k-means, k-closest neighbor, neural network, and naïve Bayes. Experimental results demonstrate that the K-closest neighbor method performs better than other models. Adnan *et al*. [5] examined three prediction models including least square support vector machine (LSSVM), classification and regression trees (CART) and group method and data handling neural network (GMDHNN) to forecast air temperature based on monthly temperature data from Pakistan. The finding was indicated that the LSSVM model is more accurate in temperature forecasting than the other two models.

Many NN models have several hyperparameters that need to adjust successfully to improve the level of accuracy of the forecasting model. Therefore, meta-heuristic algorithms are often utilized to find the best value of these hyperparameters. For example, Fei *et al*. [6] proposed particle swarm optimization-based support vector machine (PSO-SVM) to forecast grain production in India. The experimental results demonstrated that the PSO-SVM model provides better accuracy in comparison with the other two models named grey model (GM) and artificial neural network (ANN). Yasin *et al*. [7] developed a hybrid prediction technique based on grey wolf optimizer combined with least square support vector machine (GWO-LSSVM) for grid expansion and power system operation. Temperature, peak load demand, humidity and wind speed are four measured were used as input to the model. GWO was used to improve the accuracy of the proposed prediction model. Noh *et al*. [8] introduced a hybrid forecasting model that combines a genetic algorithm (GA) with gated recurrent unit (GRU). The GA-GRU model is compared with ARIMA, recurrent neural network (RNN), the standard LSTM, the standard GRU, and a hybrid GA-LSTM using a published Brazilian retailer sales dataset. The results show that the GA-GRU model dominates the other forecasting models.

This paper presents an intelligent demand forecasting model based on a hybrid of firefly algorithm and gated recurrent units (FA-GRU). The proposed model has been evaluated using historical data of 36 months of concrete block manufacturing at Dler Company in Iraq. This paper is organized: section 2 introduces the theoretical background of the proposed intelligent demand forecasting model. In section 3, simulation experiments were conducted to evaluate the performance of the proposed model using real industrial data. Finally, conclusions are summarized in section 4.

2. RESEARCH METHOD

This section presents the theoretical framework of the proposed prediction model. The proposed model for predcting the concrete block production is constracted uing gated recurrent unit deep learning method. Optimization method based on firefly algorithm is used to find the optimal hayperparametes of the proposed model which provide higer prediction accuracy.

2.1. Gated recurrent units (GRU)

GRU cell was presented by J. Noh *et al*. [9]. The GRU cell is simplified structure of LSTM cell and shows equally good learning performance [10]. Due to this simplification, the computational load has been significantly reduced which is led to an increase in the popularity of this algorithm. The structure of GRU presents in Figure 1 [11].

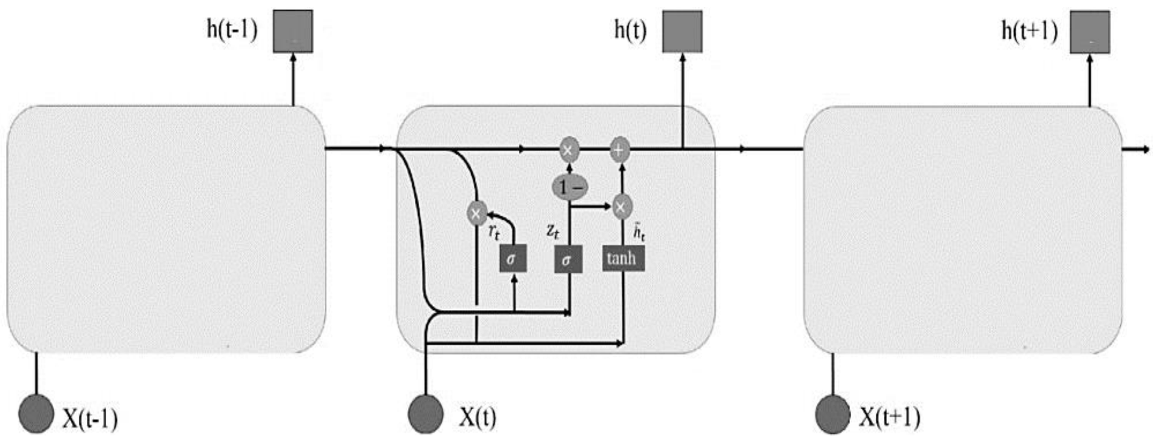

Figure 1. GRU structure

The main simplifications made are listed [12]:
- The state vectors of the LSTM are combined into a single vector $h(t)$ in the GRU.
- The entrance gate and the forget gate are controlled by a single gate controller. If the output of the gate controller is equal to 1, this refer that the forget gate is closed and the entrance gate is open. If the output is 0, it's the opposite state. So that, when memory needs to be stored, its storage location is deleted first.
- No output gate; the full state vector always comes out in step. However, which part of the previous state enters the main layer is controlled by a new gate controller.

As in a typical recurrent neural network, each GRU cell takes as input the input vector x_t and the vector c_{t-1} containing the historical information. The outputs y_t and the vectors c_t are forwarded to the next cell. Unlike a typical iterative neural network, there is a function that decides whether to update the information from the c_{t-1} vector when calculating the output of c_t. In typical iterative neural networks, the c_t vector is updated without any condition, and this increases the number of products, resulting in the disappearing gradient problem. In contrast to the LSTM [13], [14], the vectors C_t and h_t in GRU are combined in one vector as shown in (1):

$$h_t = C_t \tag{1}$$

First, the new "candidate" value of vector h_t is calculated using the input vector (x_t) and the old information vector (h_{t-1}) as shown in (2):

$$\tilde{h}_t = tanh(W_c.[h_{t-1}, x_t] + b_c) \tag{2}$$

The reason why it is a candidate value is that the calculated value is not immediately synchronized as the new information vector, as in feedforward neural networks. Instead, this candidate value is sent to a gate function, which decides whether to update the information vector with the candidate value [15]. At this stage, the tanh activation function is used as given in (3)

$$g_i = \sigma(W_i.[h_{t-1}, x_t] + b_i) \tag{3}$$

The "gate" function in (3) is the core idea behind the GRU. This gate returns a result between 0 and 1 and decides how much of the candidate h_t vector will be used in calculating the new h_t vector. In other words, it decides how much the vector, which stores the information learned from the past, will be replaced with the information learned from the new data. Thus, ensures that the vector that stores the historical information in data containing long-term relationships is kept correctly and transferred forward. The Sigmoid activation function, which returns a value between 0 and 1, is used as the activation function [16].

$$h_t = g_i * \tilde{h}_t + (1 - g_i) * h_{\,t-1} \tag{4}$$

Finally, to create the new information vector, the value obtained from the gate function is multiplied by the candidate h_t vector and summed with (1-Gate value) as shown in (4). Since the candidate value product and (1-Gate value) are added together, very large and very small numbers that may arise due to multiplication are prevented and the problem of disappearing gradient does not arise. At the same time, it is ensured that only useful information is kept in the h_t [17].

2.2. Firefly algorithm

Firefly algorithm (FA) is a swarm-based metaheuristic optimization technique proposed by Yang [18]. FA simulates the social behavior of fireflies to attract mating partners and/or to attract potential prey. The flashing light in the fireflies can be formulated as an algorithm for a function or a problem to be optimized. Yang [18] considered three assumptions for the FA which are:
- All fireflies are assumed as unisex, this means that each firefly can be attracted to other fireflies regardless of their gender.
- The attractiveness of each firefly is proportional to its brightness. For example, for any two flashing fireflies, the firefly with less bright will attract towards the brighter firefly. However, this relationship between attractiveness and brightness will be decreased as the distance between the two fireflies is increased. In this situation, firefly will move randomly.
- The brightness of a firefly is computed from the objective function.

The brightness of a firefly is computed from the objective function, and the attractiveness of each firefly is proportional to its brightness. However, the attractiveness β varies with the distance r_{ij} between the firefly i and the firefly j based on (5):

$$\beta_{ij} = \beta_o e^{-\gamma r_{ij}^2} \tag{5}$$

The distance r_{ij} between the firefly i and the firefly j is calculated based on:

$$r_{ij} = \sqrt{\sum_{k=1}^{D} (x_{i,k} - x_{j,k})^2} \tag{6}$$

where
β_{ij} Attractiveness between the firefly i and the firefly j
β_o Attractiveness at sources (i.e. $r = 0$)
γ Light absorption coefficient
D Number of decision variables in the optimization problem

The new position of the firefly i which is attracted to firefly j because firefly i has lower brightness than firefly j is calculated by:

$$x_i(t+1) = x_i(t) + \beta_o e^{-\gamma r_{ij}^2}\big(x_i(t) - x_j(t)\big) + \alpha s \tag{7}$$

where
$x_i(t+1)$ The new position of the firefly i
$x_i(t)$ The current position of the firefly i
$x_j(t)$ The current position of the firefly j
α Randomization parameter $\in [0,1]$
s Scaling factor based on the search space (i.e. upper and lower bound)

The steps of FA algorithm are shown in Figure 2.

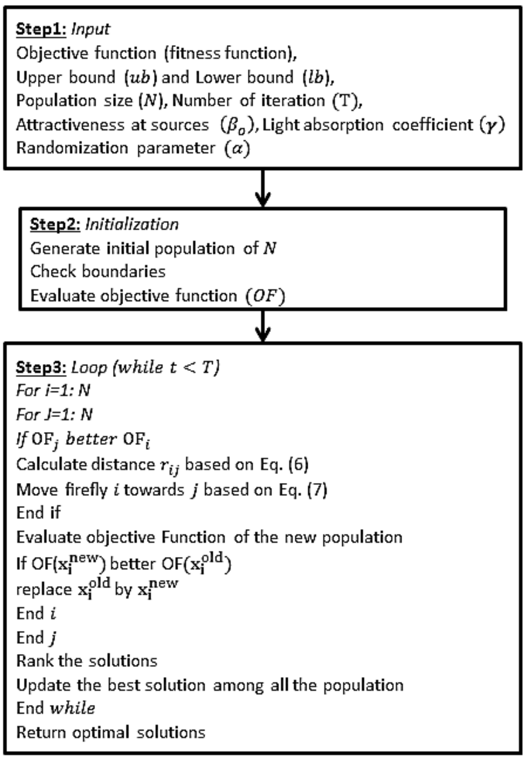

Figure 2. Flow chart of FA

2.3. Hyperparameter optimization

The parameters used while designing machine learning models are divided into two groups: parameters that can be obtained directly from the data during the training process and parameters that are predefined by the designer. These are the model parameter and the hyperparameter [19]. Model parameters are generally predicted from data. The designer is not expected to adjust these parameters. It is saved as part of the learned model. Support vectors in an SVM, weights in an ANN, coefficients in linear regression or logistic regression are some of the examples of model parameters. Unlike model parameters, hyperparameters are not estimated from the data and need to be adjusted by the designer.

Some of the hyperparameters take an infinite number of values. However, using prior knowledge about the problem helps to define a range for these values. By selecting certain main points from these specified ranges, value lists are created for hyperparameters. The architecture, arrangement, and optimization of a neural network are highly dependent on hyperparameter selection. Hyperparameter optimization (HPO) is an important component of AutoML in searching for optimal hyperparameters in the neural network structure and training process of the model. Automatic HPO facilitates fair comparisons [20]. The HPO problem has a long history dating back to the 1990 [21]. In addition, it was determined that different hyperparameter configurations were the best results for different datasets in the early stages. The number of hidden layers and the activation function, such as particle size, optimization algorithms, stochastic gradient reduction (SGD), learning rate (LR) can play an important role in determining the efficiency and accuracy of the model while it is being trained [22]. HPO can be seen as the final step in model design and the first step in training the neural network. Considering the effect of hyperparameters on accuracy and speed during training, the training process should be carefully experienced before starting [23]. The HPO process automatically optimizes the hyperparameters of the machine learning model to get humans out of the loop of the machine learning system. In this paper, the number of hidden units, batch size, learning rate, dropout rate and the epoch are five different hyperparameters in the GRU deep learning method that will be optimized using FA. The hybrid of GRU is presented in Figure 3.

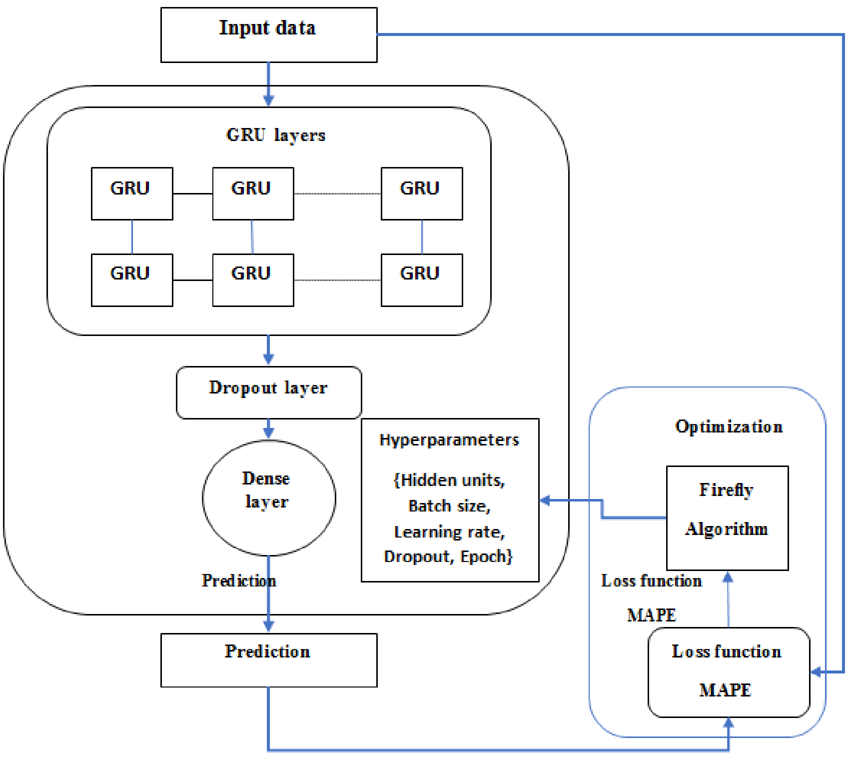

Figure 3. FA-GRU structure

3. SIMULATION EXPERIMENT

To evaluate the proposed demand forecasting model, historical data of 36 months of concrete block manufacturing at Dler Company in Iraq is used. Figure 4 presents the time series of the historical data of the sales of the company. In Figure 4, the x-axis refers to the months and the y-axis refers to the quantity in 10^3.

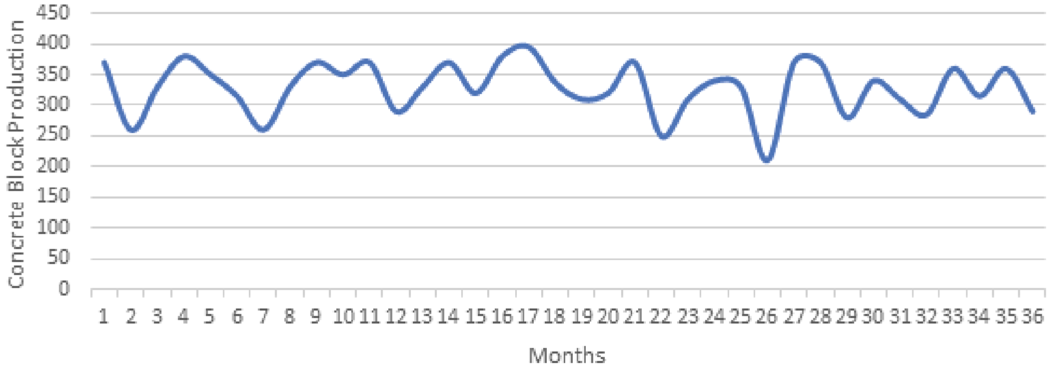

Figure 4. Historical data of the concrete block production

FA was applied to find the optimal value of the hyperparameters of the GUR based on the mean absolute percentage error (MAPE) as the objective function for the FA optimization algorithm. Table 1 provides the values of the parameters of the FA that were used in the simulation. These values are taken as recommended by [18]. The range and default values of each GRU hyperparameter are shown in Table 2. The optimal values of the hyperparameter obtained by the FA method are shown in Table 3. The default values of hyperparameters shown in Table 1 are used for training the standard GRU method while optimized values are used for the FA-GRU method.

Table 1. The parameters of FA

Parameters	Value
Population size ($\boldsymbol{N}$)	40
Attractiveness at sources ($\boldsymbol{\beta_o}$)	1
Light absorption coefficient ($\boldsymbol{\gamma}$)	1
Randomization parameter ($\boldsymbol{\alpha}$)	0.2
Number of iteration ($\boldsymbol{T}$)	10

Table 2. GRU hyperparameters

Hyperparameters	Range	Default value
Hidden units	1-250	100
Batch size	1-50	5
Learning rate	0.0001-0.02	0.001
Dropout	0.0-0.5	0.2
Epoch	5-250	100

Table 3. Optimal values of GRU hyperparameters

Hyperparameters	Optimal value
Hidden units	145
Batch size	10
Learning rate	0.002
Dropout	0.1
Epoch	112

Three metrics named root mean square error (RMSE) as given in (8), the MAPE as given in (9) and the statistical coefficient of determination (R^2) indicators as given in (10) [24], [25] were used to evaluate the performance of the proposed FA-GUR prediction model in comparison with the standard LSTM and the standard GUR.

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(A_i - F_i)^2}{n}} \tag{8}$$

$$MAPE = \sum_{i=1}^{n} \frac{|F_i - A_i|}{|F_i|} \times 100\% \tag{9}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(A_i - F_i)^2}{\sum_{i=1}^{n}(A_i - \bar{A}_i)^2} \tag{10}$$

where
i Counter (i.e. 1,2,3 …)
n Number of data
A Actual data
F Forecasted data
$\bar{A}$ Mean of the actual data

Figure 5 presents a comparison graph between the forecasted demands generated by FA-GUR and the real data for the testing data. The blue line represents the actual output, while the grey line is the predicted demand based on the proposed FA-GRU. It can be noticed that the predicted results of the testing process are closely similar to actual data. Comparison performance between the standard GUR, the standard LSTM and the proposed FA-GUR is given in Table 4. It can be observed from Table 4 that the proposed FA-GUR forecasted model archives higher accuracy in comparison with the standard GRU and standard LSTM models based on MAPE, RMSE and R^2 measures indices.

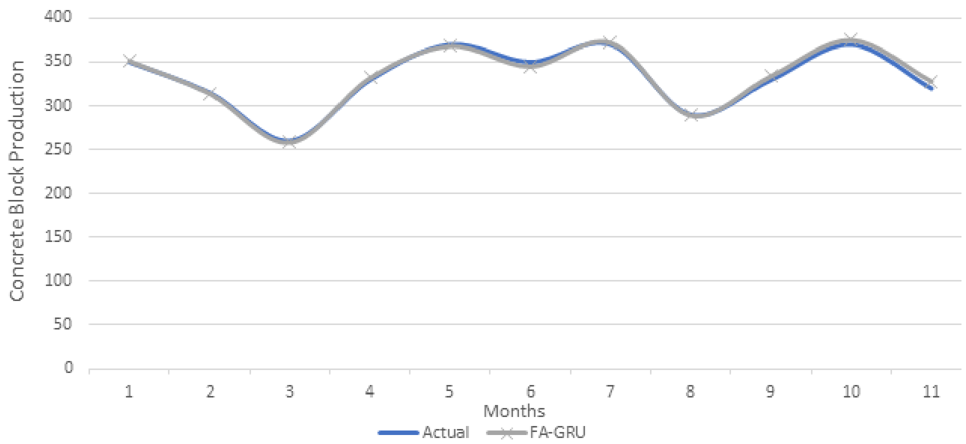

Figure 5. Comparison graph between forecasted demands generated by FA-GUR and real data

Table 4. The performance evaluation of the forecasting models

Method	MAPE	RMSE	R^2
LSTM	2.340	10.211	0.934
GRU	2.907	11.798	0.885
FA-GRU	0.900	3.680	0.990

The outcomes of the simulation experiments show that the proposed model decreases the MAPE value from 2.34 in the case of standard LSTM and 2.907 in the case of standard GRU to 0.9 for the proposed FA-GRU. In the same way, the proposed model decreases the RMSE value from 10.211 in the case of standard LSTM and 11.798 in the case of standard GRU to 3.680 for the proposed FA-GRU. Moreover, the value of R^2 increases from 0.934 in the case of standard LSTM and 0.885 in the case of standard GRU to 0.990 for the proposed FA-GRU. These results proved that the proposed FA-GRU model provides a more accurate prediction for concrete block production.

4. CONCLUSION

Demand forecasting plays an important role in production management since it has a direct impact on the profit of the company. In this paper, an intelligent demand forecasting model named GRU is proposed for the production forecasting problem. In the FA-GRU model, FA is used to select suitable parameters of GUR. The performance of the proposed prediction model was evaluated and compared with standard GRU and LSTM using historical data of 36 months of concrete block manufacturing at Dler Company in Iraq. It was observed that the proposed FA-GRU model decreases the MAPE value from 2.34 in the case of standard LSTM and 2.907 in the case of standard GRU to 0.9 for the proposed FA-GRU. In the same way, the proposed model decreases the RMSE value from 10.211 in the case of standard LSTM and 11.798 in the case of standard GRU to 3.680 for the proposed FA-GRU. Moreover, the value of R^2 increases from 0.934 in the case of standard LSTM and 0.885 in the case of standard GRU to 0.990 for the proposed FA-GRU. These results proved that the proposed FA-GRU model provides a more accurate prediction for concrete block production.

REFERENCES

[1] H. Bousqaoui, I. Slimani, and S. Achchab, "Comparative analysis of short-term demand predicting models using ARIMA and deep learning," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 11, no. 4, pp. 3319–3328, Aug. 2021, doi: 10.11591/ijece.v11i4.pp3319-3328.

[2] Z. H. Kilimci *et al.*, "An improved demand forecasting model using deep learning approach and proposed decision integration strategy for supply chain," *Complexity*, pp. 1–15, Mar. 2019, doi: 10.1155/2019/9067367.

[3] A. Utku and M. A. Akcayol, "Deep learning based prediction model for the next purchase," *Advances in Electrical and Computer Engineering*, vol. 20, no. 2, pp. 35–44, 2020, doi: 10.4316/AECE.2020.02005.

[4] D. E. Martínez Cervera, O. J. Salcedo Parra, and M. A. Aguilera Prado, "Forecasting model with machine learning in higher education ICFES exams," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 11, no. 6, pp. 5402–5410, Dec. 2021, doi: 10.11591/ijece.v11i6.pp5402-5410.

[5] R. M. Adnan *et al.*, "Air temperature prediction using different machine learning models," *Indonesian Journal of Electrical Engineering and Computer Science (IJEECS)*, vol. 22, no. 1, pp. 534–541, Apr. 2021, doi: 10.11591/ijeecs.v22.i1.pp534-541.

[6] S.-W. Fei, Y.-B. Miao, and C.-L. Liu, "Chinese grain production forecasting method based on particle swarm optimization-based support vector machine," *Recent Patents on Engineering*, vol. 3, no. 1, pp. 8–12, Jan. 2009, doi: 10.2174/187221209787259947.

[7] Z. M. Yasin, N. A. Salim, N. F. A. Aziz, Y. M. Ali, and H. Mohamad, "Long-term load forecasting using grey wolf optimizer - least-squares support vector machine," *IAES International Journal of Artificial Intelligence (IJ-AI)*, vol. 9, no. 3, pp. 417–423, Sep. 2020, doi: 10.11591/ijai.v9.i3.pp417-423.

[8] K. Cho *et al.*, "Learning phrase representations using RNN encoder-decoder for statistical machine translation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014, pp. 1724–1734, doi: 10.3115/v1/D14-1179.

[9] J. Noh, H.-J. Park, J. S. Kim, and S.-J. Hwang, "Gated recurrent unit with genetic algorithm for product demand forecasting in supply chain management," *Mathematics*, vol. 8, no. 4, Apr. 2020, doi: 10.3390/math8040565.

[10] U. Ugurlu, I. Oksuz, and O. Tas, "Electricity price forecasting using recurrent neural networks," *Energies*, vol. 11, no. 5, May 2018, doi: 10.3390/en11051255.

[11] S. Gao *et al.*, "Short-term runoff prediction with GRU and LSTM networks without requiring time step optimization during sample generation," *Journal of Hydrology*, vol. 589, Oct. 2020, doi: 10.1016/j.jhydrol.2020.125188.

[12] J. Li and J. Wang, "Forcasting of energy futures market and synchronization based on stochastic gated recurrent unit model," *Energy*, vol. 213, Dec. 2020, doi: 10.1016/j.energy.2020.118787.

[13] A. Shewalkar, D. Nyavanandi, and S. A. Ludwig, "Performance evaluation of deep neural networks applied to speech recognition: RNN, LSTM and GRU," *Journal of Artificial Intelligence and Soft Computing Research*, vol. 9, no. 4, pp. 235–245, Oct. 2019, doi: 10.2478/jaiscr-2019-0006.

[14] M. Kaselimi, N. Doulamis, A. Doulamis, A. Voulodimos, and E. Protopapadakis, "Bayesian-optimized bidirectional LSTM regression model for non-intrusive load monitoring," in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2019, pp. 2747–2751, doi: 10.1109/ICASSP.2019.8683110.

[15] R. Fu, Z. Zhang, and L. Li, "Using LSTM and GRU neural network methods for traffic flow prediction," in *2016 31st Youth Academic Annual Conference of Chinese Association of Automation (YAC)*, Nov. 2016, pp. 324–328, doi: 10.1109/YAC.2016.7804912.

[16] B. Liu, C. Fu, A. Bielefield, and Y. Liu, "Forecasting of Chinese primary energy consumption in 2021 with GRU artificial neural network," *Energies*, vol. 10, no. 10, Sep. 2017, doi: 10.3390/en10101453.

[17] K. Lee, J.-K. Kim, J. Kim, K. Hur, and H. Kim, "CNN and GRU combination scheme for bearing anomaly detection in rotating machinery health monitoring," in *2018 1st IEEE International Conference on Knowledge Innovation and Invention (ICKII)*, Jul. 2018, pp. 102–105, doi: 10.1109/ICKII.2018.8569155.

[18] X.-S. Yang, "Firefly algorithms for multimodal optimization," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 5792, 2009, pp. 169–178.

[19] J. Wu, X. Y. Chen, H. Zhang, L. D. Xiong, H. Lei, and S. H. Deng, "Hyperparameter optimization for machine learning models based on bayesian optimization," *Journal of Electronic Science and Technology*, vol. 17, no. 1, pp. 26–40, 2019, doi: 10.11989/JEST.1674-862X.80904120.

[20] A. Almalaq and J. J. Zhang, "Evolutionary deep learning-based energy consumption prediction for buildings," *IEEE Access*, vol. 7, pp. 1520–1531, 2019, doi: 10.1109/ACCESS.2018.2887023.

[21] I. Ilievski, T. Akhtar, J. Feng, and C. A. Shoemaker, "Efficient hyperparameter optimization of deep learning algorithms using deterministic RBF surrogates," in *31st AAAI Conference on Artificial Intelligence, AAAI 2017*, 2017, vol. 31, no. 1, pp. 822–829.

[22] R. Kohavi and G. H. John, "Automatic parameter selection by minimizing estimated error," in *Machine Learning Proceedings 1995*, Elsevier, 1995, pp. 304–312.

[23] N. Gorgolis, I. Hatzilygeroudis, Z. Istenes, and L. n G. Gyenne, "Hyperparameter Optimization of LSTM Network Models through Genetic Algorithm," in *2019 10th International Conference on Information, Intelligence, Systems and Applications (IISA)*, Jul. 2019, pp. 1–4, doi: 10.1109/IISA.2019.8900675.

[24] G. S. Handelman *et al.*, "Peering into the black box of artificial intelligence: evaluation metrics of machine learning methods," *American Journal of Roentgenology*, vol. 212, no. 1, pp. 38–43, Jan. 2019, doi: 10.2214/AJR.18.20224.

[25] M. Spuler, A. Sarasola-Sanz, N. Birbaumer, W. Rosenstiel, and A. Ramos-Murguialday, "Comparing metrics to evaluate performance of regression methods for decoding of neural signals," in *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, Aug. 2015, pp. 1083–1086, doi: 10.1109/EMBC.2015.7318553.

BIOGRAPHIES OF AUTHORS

Huthaifa Al-Khazraji was born in Baghdad, Iraq in 1984. He recive his B.Sc degree in Control and Systems Engineering Department at the University of Technology, Iraq in 2006. Then, the M.Sc. Degee in Industrial Engineering and Management at Politecnico di Torino University, Italy in 2010. From 2010 to 2014, he was Assistant Lecturere at Control and Systems Engineering Department at the University of Technology, Iraq. He was accepted in Central Queensland University in Australia as a Ph.D. student in March 2015. He received his Ph.D. degree in 2019. He received his scientific promotion to be a University Lecturer at Control and Systems Engineering Department in the University of Technology, Iraq in 2019. He has severals papers in modeling, simulation, control system, inventory control, process capability index, overall equipment effectiveness and assemble line balancing problem. His area of interest: control engineering, industrial engineering and optimization. He can be contacted at email: 60141@uotechnology.edu.iq.

Ahmed R. Nasser was born in Bagdad, Iraq in 1984. He got his B.Sc. degree from University of Technology Control and Systems Eng. Faculty, Baghdad Iraq in 2006, MSc degree in Computer Eng. from Istanbul University, Istanbul Turkey in 2012, Ph.D. degree from Hacettepe University, Ankara Turkey in Computer Engineering in 2018. Currently he is Lecturer at Control and Systems Engineering Department in the University of Technology. His current interest on data mining, natural language processing, and machine, and deep learning. He can be contacted at email: ahmed.r.nasser@uotechnology.edu.iq.

Sohaib Khlil is a lecturer in Applied Mechanics Engineering Department, Technical College-Baghdad, Middle Technical University, Baghdad, Iraq. Received his B.Sc. and M.Sc. in Industrial Engineering from University of Technology in Iraq. Areas of interest: planning and scheduling, quality control, time study, machine design, total productive maintenance, modeling and optimization. He can be contacted at email: sohaib-khlil@mtu.edu.iq.