Artificial Intelligence Evolution
http://ojs.wiserpub.com/index.php/AIE/

UNIVERSAL WISER PUBLISHER

Research Article

# Building Carbon Emissions Prediction Based on Deep Learning Network with Improved Particle Swarm Optimization

**Hao-Dong Chai[1], Bing-Juan Lin[1], Yi Wang[1], Yu-Hao Li[1], Ji-Yang Xu[1], Yu-Lin Lin[1], Shan-Hui Li[1], Zne-Jung Lee[2*], Yi-Zhen Chen[3]**

[1]School of Intelligent Construction, Fuzhou University of International Studies and Trade, Fujian, China
[2]School of Advanced Manufacturing, Fuzhou University, Quanzhou 362200, China
[3]Department of Communication and Technology, National Yang Ming Chiao Tung University, Hsinchu 300, Taiwan
Email: johnlee@fzu.edu.cn

**Received:** 23 August 2023; **Revised:** 13 November 2023; **Accepted:** 16 November 2023

**Abstract:** Buildings' carbon emissions are the main contributor to climate change. The world needs to be able to foresee and further reduce construction carbon emissions if it wants to prevent the worst effects of climate change. The main challenge in carbon emission prediction for buildings is how to increase algorithm accuracy. Therefore, a novel technique for calculating carbon emissions is proposed in this study. The proposed technique uses improved particle swarm optimization (PSO) and deep neural network (DNN) to anticipate carbon emissions. DNN is employed in the proposed technique to forecast carbon emissions. The parameters used are fine-tuned using the improved PSO. Additionally, it chooses features that increase the predictability of carbon emissions. In this study, many methods for predicting carbon emissions are examined, including decision tree (DT), random forest (RF), support vector regression (SVR), DNN, and the proposed method. The outcomes demonstrate that the proposed technique can lower root mean square logarithmic error (RMSLE) prediction, and the final testing result of RMLSE is 0.3476 in this study. It demonstrates that the proposed method performs better than other alternatives when compared and has good implementation ability.

***Keywords*:** carbon emissions, deep learning, particle swarm optimization, root mean square logarithmic error

## 1. Introduction

Carbon emissions have emerged as a critical signal that society, corporations, and other sectors need to comprehend and manage as nations throughout the world pay increasing attention to global climate change. Buildings are one of the largest sources of energy consumption in many nations since they serve as the primary spatial carrier for people's lives and activities. Large-scale new building construction and the operation of enormous old buildings have produced a significant amount of carbon dioxide emissions in recent years due to the rapid rise of urban construction [1]. Cuéllar-Franca et al. [2] have studied three types of buildings common in the UK, considering the environmental impact of each stage from a full life cycle perspective. The results show that the greenhouse gas emissions in the building use stage have the greatest impact on the environment, and the recycling and utilization of building materials are conducive to reducing greenhouse gas emissions. Suzuki et al. [3] conducted research on residential buildings in Japan, used the industrial


DOI: https://doi.org/10.37256/aie.4220233561

balance sheet method to calculate the carbon emissions and energy consumption of residential buildings throughout the life cycle, and proposed an input-output calculation method for calculating $CO_2$ emissions for the whole process of building construction, use, maintenance, demolition, and cleanup. Nassen et al. [4] used the top-down input-output analysis method to calculate the energy consumption and $CO_2$ emissions in the construction stage of a Swedish building, and compared the results with the case results calculated by the bottom-up full life cycle method, and found that the calculation results of the input-output method were 90% higher than that of the life cycle method. Carbon dioxide ($CO_2$) emissions from building operations peaked in 2021, according to the 2022 Global Building and Construction Industry Status Report. The construction industry's goal of achieving ultra-low emissions or perhaps zero emissions is a crucial first step toward achieving carbon peak and carbon neutrality as a response to the increasingly catastrophic situation of building carbon emissions. The real estate building business is currently supporting carbon reduction efforts in a progressive manner, but there are still numerous obstacles to overcome, including inconsistent norms, delayed actual development, and expensive new technology. The following variables are the main causes of the distortion of carbon emission data. First off, the results from carbon accounting are not accurate enough. Currently, the majority of carbon accounting research is based on IPCC accounting methodologies and guidelines [5], but this methodology is less precise and useful due to regional differences. The limits of carbon accounting guidelines are also becoming hazy. As a result, it has always been challenging to anticipate carbon emissions accurately and effectively. Machine learning has recently shown promise in helping to implement a number of carbon emissions policies, including carbon peaking and carbon neutrality. The traditional building carbon emissions prediction sector is evolving in an intelligent manner, propelled by the new generation of information technology. This branch of study has gradually advanced and become more in-depth.

Machine learning (ML) has been utilized to resolve this issue and forecast the $CO_2$ concentration in order to test and monitor $CO_2$ [6-13]. In order to forecast the findings of $CH_4$ and $CO_2$ concentrations in deltaic water bodies, Ludwig et al. [14] constructed a model utilizing machine learning. However, the majority of research focuses on machine learning for predicting carbon emissions, whereas feature extraction for predicting carbon emissions receives less attention. To estimate carbon emissions, Lee et al. [15] proposed using deep learning with autonomous feature engineering; however, parameters still need to be fine-tuned to increase prediction accuracy. In this research, we propose a DNN-based building carbon emissions prediction with improved PSO. The proposed approach makes efficient use of the improved PSO to select features and perform fine-tuning of parameters to increase the predictability of carbon emissions.

The remainder of this essay is structured as follows. The related methods of DT, RF, SVR, PSO, and DNN are introduced in Section 2. Described in Section 3 is the proposed algorithm. The outcomes are shown in Section 4. Finally, Section 5 provides the conclusions.

## 2. The brief description of DT, RF, SVR, PSO, and DNN

In this section, the related approaches of the decision tree, random forest, support vector regression, particle swarm optimization, and deep learning network are introduced.

### 2.1 *Decision tree*

The procedure used to choose the optimum splitting feature for each node in the DT algorithm is known as feature selection [16]. To create a corresponding feature branch for each value and divide the related data set samples, the most often used feature selection is condensed. Numerous subsets can be produced by such a partition. The data set will show more nodes as a result of the data set's continuous division; this is the Gini coefficient, and equation (1) can be used to calculate it.

$$Gini(t) = 1 - \sum \left( p(i|t)^2 \right) \tag{1}$$

where $t$ represents a specific node, $i$ represents the index of the class, and $p(i|t)$ represents the proportion of samples

belonging to a class in a node; the construction of a decision tree is the process of growing a tree by recursively partitioning the nodes.

## 2.2 *Random forest*

As depicted in Figure 1, the RF is an ensemble learning technique based on decision trees that excels at processing labeled data sets [17]. The data set is separated into several subsets using the bagging algorithm formula, and a decision tree is built for training for each subset. The principles of feature selection and node splitting are the foundations upon which the decision tree is built. The best features are chosen at each node to execute splitting, typically with the goal of lowering impurity. Until the termination condition is met, this recursive process keeps running. Each decision tree makes an individual forecast during the prediction stage, and the ultimate result is then decided by voting or averaging. It averages the regression tree's outcomes if the issue is one of regression. The RF is popular in machine learning because of its many benefits. It can manage big data sets and high-dimensional feature spaces, for starters. Random forests are successful against the dimensionality curse because each decision tree is trained on just a small subset of features. It is additionally resistant to missing numbers and outliers. The impact of missing data and outliers is minimal because each decision tree is trained using partial samples and features. Because of this, random forests are excellent for handling erratic or insufficient real-world data.

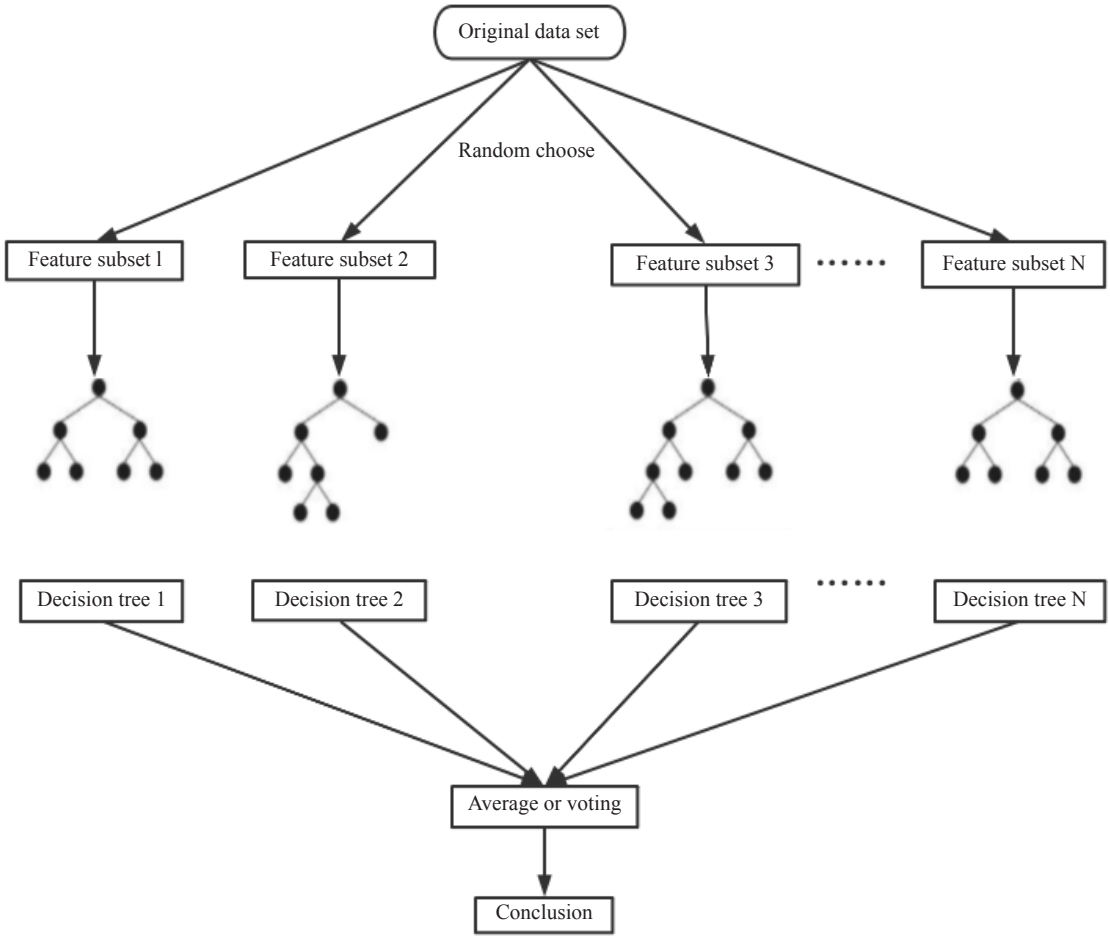


**Figure 1.** Random Forest

## 2.3 *Support vector regression*

A machine learning algorithm addressing regression issues is the SVR [18]. SVR provides particular advantages and features in conventional regression problems compared to conventional regression techniques. The fundamental tenet of the SVR method is that the model can be expressed by equation (2) by identifying an ideal separating hyperplane.

$$f(x) = w^T \Phi(x) + b \tag{2}$$

where $w$ is the model parameter, $b$ is a model parameter which means that the sample points are mapped to a high-dimensional space, and the error between the predicted value and the real value is minimized. Support vectors are the training sample points that are closest to the hyperplane in SVR. The construction of the hyperplane and the prediction process both heavily rely on these support vectors. Unlike the support vector used in classification, the support vector used in SVR can be in the error band. The distance of the support vector from the hyperplane, which is determined by the loss function as shown in equation (3), defines the error between the forecast result and the true value.

$$L(\omega, b) = C \sum \xi + 1/2 \|\omega\|^2 \tag{3}$$

where $||\omega||^2$ represents the $L_2$ norm of the weight vector $\omega$, $\xi$ is the slack variable, and $C$ is a regularization parameter that controls the complexity of the model.

## 2.4 *Particle swarm optimization*

The PSO is regarded as one of the modern innovative heuristic algorithms because of how widely used it is and how easily it can be implemented [19]. The PSO algorithm has carefully researched the collective behavior of animals with an eye to the future to serve as a dependable method to solve optimization issues in a wide range of applications. In PSO, each possible solution is represented by a particle, and the population of the algorithm is represented by the entire swarm. The particles are able to learn and develop to the highest level of efficiency because of the swarm's collaboration and information sharing. For $n$-dimension, each particle of the swarm is represented by $x_{ij} = (x_{i1}, x_{i2}, \ldots, x_{ij})$, and the position is as shown in equation (4).

$$x_{ij}(t) = x_{ij}(t-1) + v_{ij}(t) \quad i, j = 1, 2, \cdots, n \tag{4}$$

where $x_{ij}(t)$ is the current position, $x_{ij}(t-1)$ is the previous position and $v_{ij}(t)$ is the velocity which determines the movement of each particle in the current iteration. The velocity of the particle is given by the following equation (5).

$$v_{ij}(t) = w v_{ij}(t-1) + \alpha r_1 \left[ pbest_{ij}(t-1) - x_{ij}(t-1) \right] + \beta r_2 \left[ gbest_{ij}(t-1) - x_{ij}(t-1) \right], \; i, j = 1, 2, \cdots, n \tag{5}$$

Where $v_{ij}(t)$ denotes the velocity in the current iteration, $v_{ij}(t-1)$ is the velocity in the previous iteration, $w$ is the inertia weight, $r_1$ and $r_2$ are two variables which are randomly derived from uniform distribution in range [0, 1], acceleration coefficients with a significant impact on the effectiveness of the PSO method are defined as $\alpha$ and $\beta$, $pbest(t-1)$ is the best position of the particle up to the previous iteration, and $gbest_{ij}(t-1)$ is the best position of the entire swarm up until the same iteration. As a form of memory, it stores the best prior location that the particle has attained as $[pbest_{ij}(t-1) - x_{ij}(t-1)]$. The term $[gbest_{ij}(t-1) - x_{ij}(t-1)]$ refers to the particles that act in accordance with the information they have learned from the swarm, using the swarm's optimal position as a guide.

## 2.5 *Deep learning network*

A strong nonlinear modeling tool is the DNN, an artificial neural network having several layers between the input and output layers [20]. Figure 2 depicts the DNN's fundamental structure. A neural network has two hidden layers, and each layer's neurons are connected to those in the layer before it. Each neuron processes information from the layer of neurons above it using weights and activation functions before generating an output. Equation (6) describes the output of neuron $j$.

$$y_j = f\left(\sum_{i=1}^{n} w_{ij} x_i - \theta_j\right) \tag{6}$$

where $y_j$ is the output of $j$th neuron, $x_j$ is the $i$th input, $w_{ij}$ is the weight for the $i$th input and $j$th neuron, $n$ is the total number of input, $\theta_j$ is the threshold of $j$th neuron, and $f$ is the activation function.

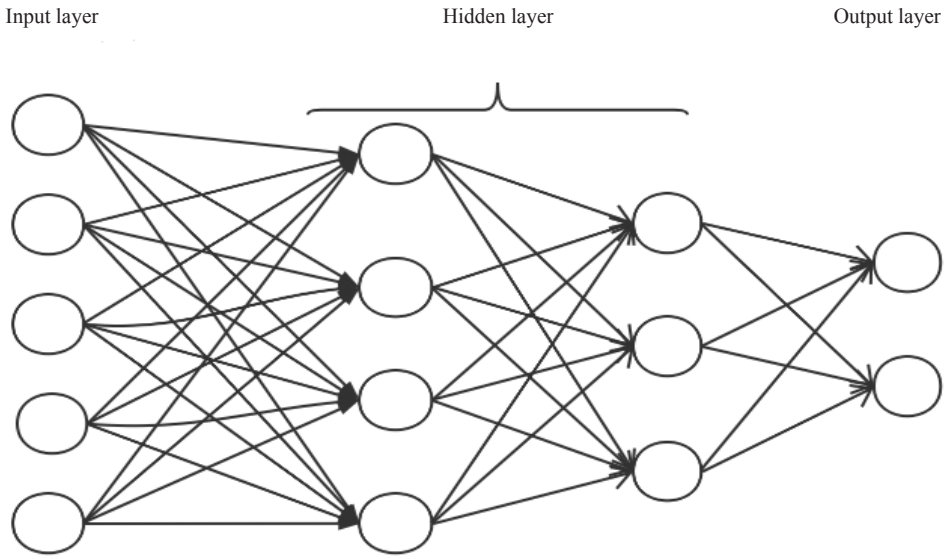


**Figure 2.** The deep learning network structure

# 3. The Proposed method

The development of carbon emission predictions using deep learning and enhanced particle swarm optimization is proposed in this research. Figure 3 displays the proposed algorithm's flowchart. The used data for carbon emissions are initially loaded into Figure 3. The DNN is then used to make the best possible prediction of carbon emissions. The enhanced PSO is then used to carry out feature selection and parameter fine-tuning. The proposed procedure is then carried out until the stop requirements are satisfied. This study's carbon emission dataset was downloaded from Kaggle with ten features [21]. Table 1 provides a thorough explanation of these ten features.

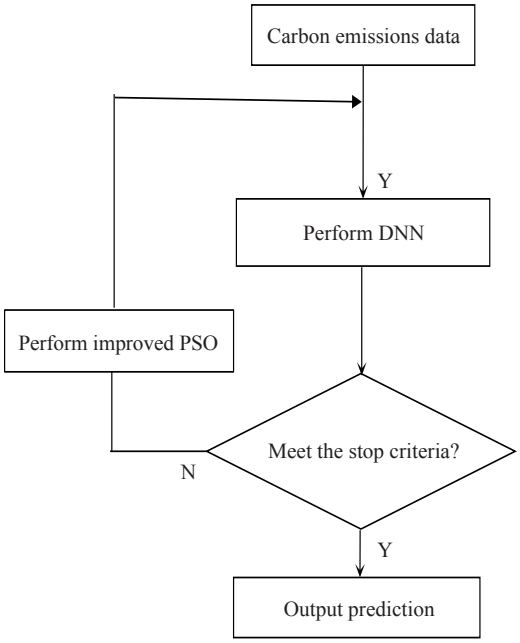


**Figure 3.** The flowchart of the proposed method

**Table 1.** The used carbon emissions dataset

| # | Characteristics |
|---|---|
| 1 | Year |
| 2 | Energy Related $CO_2$ Emissions |
| 3 | Oil Production |
| 4 | Natural Gas Production |
| 5 | Coal Production |
| 6 | Electricity Generation |
| 7 | Hydroelectricity Usage |
| 8 | Nuclear Energy Usage |
| 9 | Installed Solar Capacity |
| 10 | Installed Wind Capacity |

The structure of DNN is set up as shown in Table 2. In Table 2, dense_input is the input layer with 1,024 neurons. The dense_1, dense_2, and dense_3 are hidden layers. The dense_output is output layer. The activation function of the input layer and the hidden layers are set as relu function. For the output layer, the activation is set to linear function.

For improved PSO, the value of weight $w$ varies according to the number of the current iteration $t$ and the number of maximum iterations $t_{max}$ = 5,000. It is described as follows in equation (7).

$$w = \frac{t_{\max} - t}{t_{\max}} \tag{7}$$

The fitness value of the $j$th particle is shown as equation (8).

$$Fitness_j = 1/\left(1 + RMLSE_j\right) \tag{8}$$

The following is the basic process of the improved PSO algorithm.
Step 1: Generate starting particles in the population at random.
Step 2: Using equation (8), calculate the fitness value of each particle in the population.
Step 3: Using equation (7), calculate the weight $w$.
Step 4: Using equation (5), calculate the velocity of each particle.
Step 5: Using equation (4), move each particle in the population to the next place.
Step 6: If the stop criteria is met, the algorithm is terminated. Otherwise, proceed to Step 2.

**Table 2.** The structure of DNN

| Layer | Shape |
|---|---|
| input | 1,024 |
| hidden_1 | 512 |
| hidden_2 | 256 |
| hidden_3 | 128 |
| output | 1 |

## 4. Simulation results

For the simulation, we defined the stop condition as the output prediction that has not been improved further after 30 iterations. In this study, 80% of the data on carbon emissions is utilized as training data, while the remaining data is used as testing data. The representation of a particle's solution is specified as the values of $\alpha$, $\beta$, $\theta_j$, and the number of 10 features in the employed carbon emissions dataset for feature selection to apply the improved PSO. The proposed technique requires 10 features from the carbon emissions dataset to determine which features are chosen. The value of characteristics ranges from 0 to 1. If a feature's value is less than or equal to 0.5, the corresponding feature is not chosen. If, on the other hand, a feature's value is larger than 0.5, the corresponding feature is chosen. Table 3 shows the outcomes of various methodologies for predicting carbon emissions. It should be noted that the proposed method and the DNN have the same architecture. The stop condition for SVR, DNN, and the proposed method are all the same condition. It is demonstrated that the proposed method has the lowest RMLSE, and the result is shown in Figure 4. Furthermore, the proposed method includes four features: oil production, coal production, hydroelectricity usage, and installed solar capacity.

**Table 3.** The results of various approaches to carbon emissions

| Approaches | RMLSE |
|---|---|
| DT | 0.7597 |
| RF | 0.5355 |
| SVR | 1.7748 |
| DNN | 0.4973 |
| The proposed method | 0.3476 |

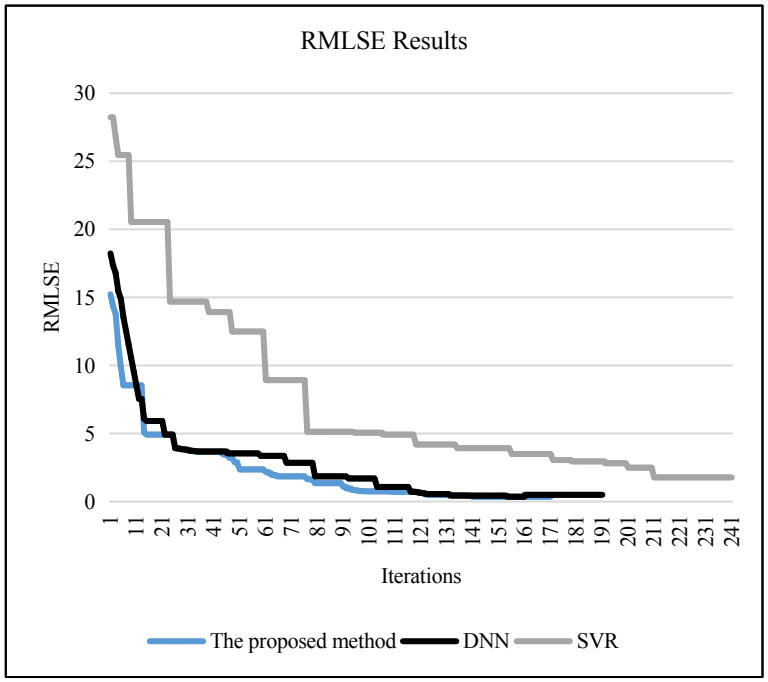


**Figure 4.** The result of RMLSE

## 5. Conclusions

This paper proposes deep learning-based carbon emission prediction with improved PSO. The proposed technique employs improved PSO for feature selection and parameter fine tuning. Carbon emissions are predicted using a deep learning network. According to the results, the proposed method has the lowest RMLSE of the techniques tested. It is also discovered that the improved PSO features oil production, coal production, hydropower usage, and installed solar capacity. In fact, the proposed method beats all previous algorithms tested. For carbon emission prediction, it could depend on the factor of seasons. The long short-term memory (LSTM) and gate recurrent unit (GRU) are time-recurrent neural network (RNN). They are proposed to solve problems such as long-term memory and gradients in backpropagation. In future work, these approaches may do more research for predicting carbon emissions with the factor of seasons.

## Acknowledgment

This paper was supported by the Fuzhou University of International Studies and Trade.

## Conflict of interest

The authors declare no competing financial interest.

## References

[1] Zhang Z, Xie H, Zhang J, Wang X, Wei J, Quan X. Prediction and trend analysis of regional industrial carbon emission in China: A study of Nanjing city. *International Journal of Environmental Research and Public Health*. 2022; 19(12): 7165.
[2] Cuéllar-Franca RM, Azapagic A. Environmental impacts of the UK residential sector: Life cycle assessment of houses. *Building and Environment*. 2012; 54: 86-99.
[3] Suzuki M, Oka T, Okada K. The estimation of energy consumption and $CO_2$ emission due to housing construction in Japan. *Energy and Buildings*. 1995; 22(2): 165-169.
[4] Nässén J, Holmberg J, Wadeskog A, Nyman M. Direct and indirect energy use and carbon emissions in the production phase of buildings: An input-output analysis. *Energy*. 2007; 32(9): 1593-1602.
[5] Yin L, Sharifi A, Huang LQ, Chen JY. Urban carbon accounting: An overview. *Urban Climate*. 2022; 44: 101195.
[6] Kapoor NR, Kumar A, Kumar A, Kumar A, Mohammed MA, Kumar K, et al. Machine learning-based $CO_2$ prediction for office room: A pilot study. *Wireless Communications and Mobile Computing*. 2022; 2022: 1-16. Available from: https://doi.org/10.1155/2022/9404807.
[7] Kumari S, Singh SK. Machine learning-based time series models for effective $CO_2$ emission prediction in India. *Environmental Science and Pollution Research*. 2022; 1-16. Available from: https://doi.org/10.1007/s11356-022-21723-8.
[8] Gupta S, Li L. The potential of machine learning for enhancing $CO_2$ sequestration, storage, transportation, and utilization-based processes: A brief perspective. *JOM*. 2022; 74(2): 414-428.
[9] Ağbulut Ü. Forecasting of transportation-related energy demand and $CO_2$ emissions in Turkey with different machine learning algorithms. *Sustainable Production and Consumption*. 2022; 29: 141-157.
[10] Kuroki N, Suzuki Y, Kodama D, Chowdhury FA, Yamada H, Mori H. Machine learning-boosted design of ionic liquids for $CO_2$ absorption and experimental verification. *The Journal of Physical Chemistry B*. 2023; 127(9): 2022-2027.
[11] Lok JY, Tsai WH, Cheng IC. A hybrid machine learning-genetic algorithm (ML-GA) model to predict optimal process parameters of nanoporous Cu for $CO_2$ reduction. *Materials Today Energy*. 2023; 36: 101352.
[12] Yılmaz B, Oral B, Yıldırım R. Machine learning analysis of catalytic $CO_2$ methanation. *International Journal of Hydrogen Energy*. 2023; 48(64): 24904-24914.
[13] Lee ZJ, Lin Y, Chen ZY, Yang Z, Fang WG, Lee CH. Ensemble deep learning applied to predict building energy consumption. *2023 IEEE 6th Eurasian Conference on Educational Innovation (ECEI)*. Singapore: IEEE; 2023. p.339-341.
[14] Ludwig SM, Natali SM, Mann PJ, Schade JD, Holmes RM, Powell M, et al. Using machine learning to predict inland aquatic $CO_2$ and $CH_4$ concentrations and the effects of wildfires in the Yukon-Kuskokwim Delta, Alaska. *Global Biogeochemical Cycles*. 2022; 36(4): e2021GB007146.
[15] Lee ZJ, Lin Y, Yang Z, Chen ZY, Fan WG, Lee CH. Novel automatic feature engineering for carbon emissions prediction base on deep learning. *2023 IEEE 6th Eurasian Conference on Educational Innovation (ECEI)*. Singapore: IEEE; 2023. p.203-206
[16] He HM, Chen Y, Xiao JY, Chen XQ, Lee ZJ. Data analysis on the influencing factors of the real estate price. *Artificial Intelligence Evolution*. 2021; 2(2): 52-66.
[17] Belgiu M, Drăguţ L. Random forest in remote sensing: A review of applications and future directions. *ISPRS Journal of Photogrammetry and Remote Sensing*. 2016; 114: 24-31.
[18] Smola AJ, Schölkopf B. A tutorial on support vector regression. *Statistics and Computing*. 2004; 14: 199-222.
[19] Lin SW, Ying KC, Chen SC, Lee ZJ. Particle swarm optimization for parameter determination and feature selection

of support vector machines. *Expert Systems with Applications*. 2008; 35(4): 1817-1824.
[20] Shi LF, Liu ZY, Zhou KJ, Shi Y, Jing X. Novel deep learning network for gait recognition using multimodal inertial sensors. *Sensors*. 2023; 23(2): 849.
[21] https://www.kaggle.com/datasets/iannjuguna/energy-data-since-1990 [Accessed 1st September 2023].