Prediction Model for Carbon Emissions in Hebei Province: An Empirical Study Based on Boosting Algorithm

Yiming Huo *

Department of Electrical Engineering and Automation, Hebei University, Baoding, China, 071002

* Corresponding Author Email: hbdxhym@126.com

Abstract. In light of China's ambitious "dual carbon" targets, accurate emission predictions are essential for effective reduction strategies. Traditional forecasting methods often fail to capture the complex nonlinear relationships in emission data, whereas Boosting algorithms enhance prediction accuracy by addressing these challenges. This study leverages Boosting algorithms, a state-of-the-art ensemble learning technique, to predict future carbon emissions in Hebei. By incorporating critical factors such as industrial output, energy consumption, and population dynamics, the study addresses the nonlinearities inherent in carbon emission data, which traditional forecasting models like ARIMA (Autoregressive Integrated Moving Average) and RF (Random Forest) struggle to capture. The results demonstrate that the Boosting model significantly outperforms conventional approaches, yielding the most accurate and robust predictions. This research offers valuable insights for shaping targeted carbon reduction strategies in Hebei, supporting its efforts to meet its carbon peak target, and contributing to China's broader climate objectives. The findings underscore the potential of advanced machine learning techniques to inform data-driven decision-making in the pursuit of sustainable development and carbon neutrality.

Keywords: Hebei Province, Boosting Algorithm, Carbon Emissions.

1. Introduction

Climate change represents one of the most urgent and complex challenges facing humanity today, with widespread implications for ecosystems, public health, and global economic stability. As part of a concerted global effort to mitigate the effects of climate change, reducing greenhouse gas (GHG) emissions has become a fundamental objective for nations worldwide [1]. In particular, China, as the world's largest emitter of carbon dioxide, has set bold and ambitious climate goals under its "dual carbon" strategy. These include the goal of peaking carbon emissions by 2030 and achieving carbon neutrality by 2060 [2]. Meeting these targets requires comprehensive policy frameworks and a concerted effort at the regional level, where local governments play a critical role in emission reduction and energy transition efforts. Hebei Province, an industrial powerhouse in northern China, faces particular challenges in this regard [3]. The province has one of the highest levels of carbon emissions in the country, driven primarily by its dependence on coal and its concentration of energy-intensive industries such as steel, cement, and chemical manufacturing. With an energy structure largely reliant on fossil fuels, Hebei confronts considerable difficulties in reducing emissions while maintaining industrial growth and economic stability [4]. As such, the province is at the forefront of China's efforts to balance industrial development with the need for environmental sustainability.

To effectively reduce emissions and support the transition to a low-carbon economy, it is essential to have reliable, accurate forecasts of future carbon emissions. Such forecasts serve as the foundation for designing targeted emission reduction policies, optimizing energy systems, and transitioning to cleaner industrial practices [5]. Traditional forecasting methods, such as linear regression and time series models like ARIMA [6], have been widely used for emission prediction. However, these approaches often struggle to capture the complex, nonlinear relationships among the various factors influencing carbon emissions, such as industrial output, energy consumption, and technological advancements. In recent years, machine learning (ML) methods [7] have shown great promise in overcoming the limitations of traditional forecasting techniques. These methods excel at modeling nonlinear relationships and can handle large, complex datasets with multiple variables, making them

ideal for carbon emission prediction. Among the ML techniques, boosting algorithms [8] have gained significant attention for their ability to improve prediction accuracy and model stability. By combining multiple weak models into a single, stronger model, boosting algorithms enhance the reliability of predictions, which is crucial for informing emission reduction strategies and policy decisions.

This study aims to apply Boosting algorithms to forecast carbon emissions in Hebei Province, with a focus on identifying the key factors that influence emissions in this industrialized region. By incorporating variables such as energy consumption, industrial activity, and policy interventions, this research seeks to provide a comprehensive analysis of carbon emission trends in Hebei. The ultimate goal is to offer valuable insights and data-driven recommendations that will support the development of effective carbon reduction strategies, assisting Hebei in meeting its carbon peak target and contributing to China's broader climate objectives.

2. The basic fundamentals of Boosting algorithms

2.1. The structure of Boosting algorithms

Boosting is an ensemble learning technique that combines multiple weak learners to create a strong model [9]. The basic idea is to improve the accuracy of predictions by sequentially applying weak models and focusing on the mistakes made by previous models. The algorithms' structure is shown in Figure 1.

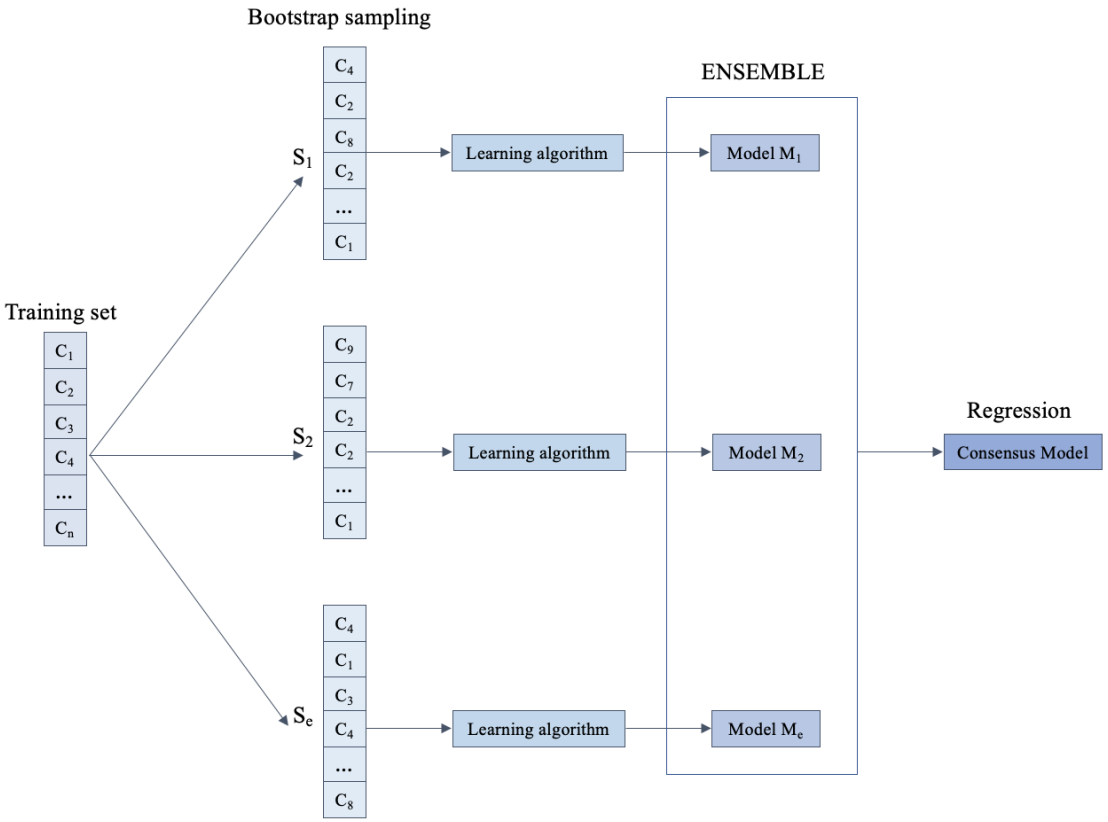

Figure 1. Boosting Framework Diagram

The general model of Boosting forecast consists of four basic elements, which are:

(1) Weak Learners: Boosting works by combining weak learners, which are models that perform slightly better than random guessing [10]. These are typically decision trees with a small depth, known as stumps. The strength of the final model comes from the combination of these simple learners.

(2) Sequential Learning: In Boosting, the models are trained sequentially, meaning each new model is trained to correct the errors made by the previous one. This is done by assigning higher weights to the data points that were misclassified by earlier models, forcing the next model to focus more on those difficult cases.

(3) Weighted Data Points: Initially, each training data point has equal weight. As the algorithm progresses, weights are adjusted based on the errors. If a data point was misclassified by a previous model, its weight increases, making it more important for the next model.

(4) Model Combination: After all the weak learners are trained, the predictions from each model are combined. The final prediction is made based on a weighted vote (for classification) or weighted average (for regression) of the predictions from each individual model. Models that perform better on the training data will have higher weights in the final combination.

The objective of the weak learner in round t is to minimize the weighted error rate can be defined as follow:

$$h_t(x) \approx \arg\min_h \sum_{i=1}^{N} D_t(i) \cdot \mathbf{II}(y_i \neq h(x_i)) \tag{1}$$

$$\varepsilon_t = \frac{\sum_{i=1}^{N} D_t(i) \cdot \mathbf{II}(y_i \neq h_t(x_i))}{\sum_{i=1}^{N} D_t(i)} \tag{2}$$

Where x represents the input data; $h_t(x)$ represents the prediction output of the weak learner; N represents the total number of samples; $D_t(i)$ represents the weight of the sample i in round t; $\mathbf{II}(\cdot)$ represents the indicator function (equal to 1 if $y_i \neq h(x_i)$, otherwise 0); ε_t represents the weighted error rate of the weak learner in round t.

Then, calculate the weight of the weak learner based on the error rate α_t. The weight is typically determined based on the error rate, with the formula (3).

$$\alpha_t = \frac{1}{2}\ln(\frac{1-\varepsilon_t}{\varepsilon_t}) \tag{3}$$

If ε_t is small, it indicates that the weak learner performs well, and α_t will be relatively large. Conversely, if ε_t is large, it indicates poor performance of the model, and α_t will be relatively small.

After each round, Boosting adjusts the weights of the data points based on the performance of the weak learner. The weight update formula for the samples is:

$$D_{t+1}(i) = D_t(i) \cdot \exp(-\alpha_t \cdot y_i \cdot h_t(x_i)) \tag{4}$$

Where $D_t(i)$ represents the weight of sample i in round t; α_t represents the weight of the weak learner in round t; y_i represents the true label of sample i, usually equal to ± 1; $h_t(x_i)$ represents the prediction of the weak learner for sample i in round t.

Next, normalize the weights of all samples so that their sum is 1, maintaining consistency in the data distribution:

$$D_{t+1}(i) = \frac{D_{t+1}(i)}{\sum_{i=1}^{N} D_{t+1}(i)} \tag{5}$$

The key idea of Boosting is to combine the results of multiple weak learners into a strong learner through weighted aggregation. The final model prediction $F(x)$ is the weighted sum of the predictions of all weak learners, given by:

$$F(x)=\sum_{t=1}^{T}\alpha_t \cdot h_t(x) \tag{6}$$

2.2. Input Feature Handling for Boosting Prediction Model

Boosting algorithms, is known for its iterative nature, where each new model corrects the errors of the previous ones. While this approach is powerful, it can also become computationally expensive, especially when the input feature space is large. Principal Component Analysis (PCA) can help mitigate this by transforming the original features into a lower-dimensional space, where the most significant features (principal components) are retained, and the less informative ones are discarded [11]. The PCA structure is shown in Figure 2.

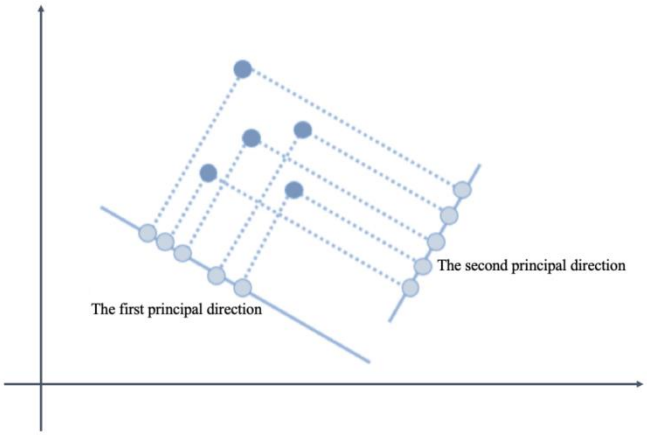

Figure 2. The structure of PCA

The process of data handling in PCA includes the following steps:

(1) Data Standardization: Each feature is centered by subtracting its mean, ensuring a mean of 0 for each feature. Then, each feature is standardized by dividing by its standard deviation, making the standard deviation 1.

(2) Covariance Matrix Calculation: The covariance matrix is computed, and then eigenvalue decomposition is applied to obtain the eigenvalues and corresponding eigenvectors. The eigenvalues and eigenvectors are sorted in descending order based on the eigenvalues. The number of principal components retained is typically determined by the cumulative contribution rate of the eigenvalues.

The standardized data $X_{m,j}$ can be represented as:

$$X_{m,j}=\frac{x_{m,j-\mu_j}}{\sigma_j} \tag{7}$$

Where $x_{m,j}$ represents the carbon emission of a city in year m for the j-th feature; μ_j and σ_j represents the mean and standard deviation of this feature from 2000 to 2021.

The corresponding covariance matrix is represented as:

$$C=(1/c)\cdot M\cdot M^T \tag{8}$$

Where C represents the covariance matrix of the relationships between different features in the data; $c_{u,v}$ represents the elements of the covariance matrix, which also means the covariance between feature u and feature v; M^T represents the transpose of M.

The reduced-dimensionality dataset O is:

$$O = M \square W \tag{9}$$

Where W represents a matrix containing the first w eigenvectors.

2.3. Model Selection and Comparison

In order to accurately predict carbon emissions in Hebei Province, two commonly used prediction models were selected for comparison:

ARIMA Model [12]: The ARIMA model is a traditional time series analysis method that is widely used for data with trend and seasonality. This model assumes that the data follows a certain linear pattern, making it suitable for relatively stable time series data.

RF Model [13]: The RF model is an ensemble learning method that builds multiple decision trees and combines their outputs to increase prediction accuracy. Due to its ability to effectively capture nonlinear relationships, it is well-suited for complex carbon emission data.

2.4. Evaluation Methods for Similar Models

To test the performance of the forecasting model, this paper introduces MAE (mean absolute error), root RMSE (means square error) [14], which can be calculated by (10)-(11)

$$\mathrm{MAE} = \frac{1}{\mathrm{A}} \sum_{a=1}^{\mathrm{A}} \left| f_{emission,a}^{potential} - f_{emission,a}^{actual} \right| \tag{10}$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{\mathrm{A}} \sum_{a=1}^{\mathrm{A}} \left(f_{emission,a}^{potential} - f_{emission,a}^{actual} \right)^2} \tag{11}$$

Where A represents the sample size; $f_{emission,a}^{actual}$ and $f_{emission,a}^{potential}$ represent the actual and predicted carbon emission values for a certain city in year a.

RMSE is a widely used metric for assessing the prediction error of regression models. It calculates the mean of the squared differences between predicted values and actual values, then takes the square root. RMSE is sensitive to larger errors, making it effective in highlighting large prediction deviations. MAE measures the average absolute difference between predicted and actual values. Unlike RMSE, MAE is less sensitive to outliers and provides a more balanced measure of prediction performance across all samples.

3. Results

3.1. Dataset and Parameter Settings

The data used in this study primarily comes from various yearbooks and statistical materials published by national statistical departments [15]. This includes carbon emission data from 11 cities in Hebei Province from 2000 to 2021, with yearly intervals. The categories of carbon emissions covered in the data include emissions from urban transportation and construction, industrial production processes, agriculture, forestry, and land-use changes, waste treatment, electricity, heating, and cooling purchased to meet urban consumption, and emissions associated with the production, transportation, use, and waste treatment of all goods purchased from outside the city for consumption, among others. The overall carbon emissions for each of the 11 cities are shown in Figure 3.

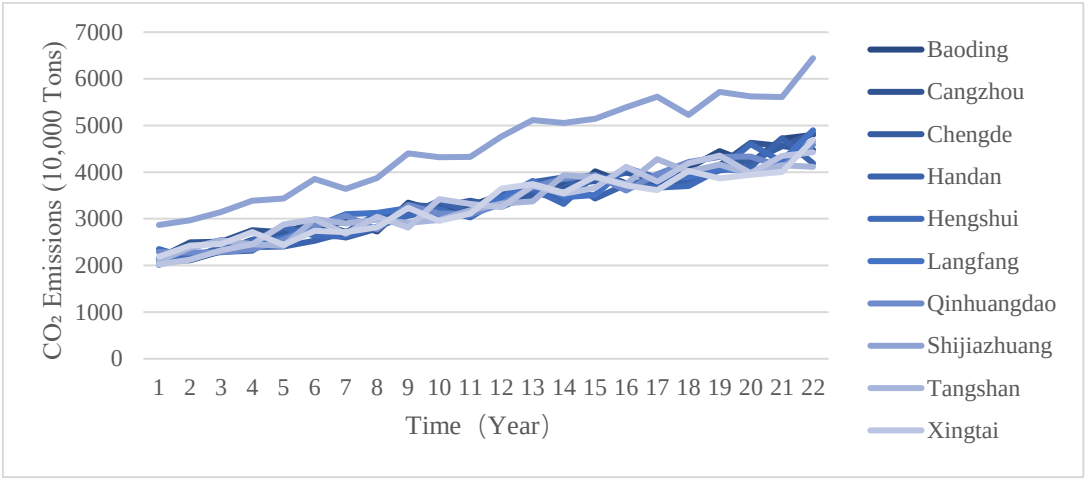

Figure 3. The overall carbon emissions of 11 cities

Our models are developed using TensorFlow, Google's deep learning library for Python. All experiments are conducted on a desktop workstation powered by an Intel Core i9-13900K processor, equipped with 32 GB of RAM and an NVIDIA GeForce RTX 4090 GPU. The parameter of the Boosting model used in this paper is listed in Table 1.

Table 1. Parameter setting of the Boosting model

Parameters	Value
Learning rate	0.01
Gamma	2
Subsample	0.8
Estimators	320
Booster	gbtree
Tree method	auto

3.2. Analysis of the input feature loadings

In this study, PCA was utilized to analyze the main input features affecting carbon emissions in Hebei Province. By reducing the dimensionality of the dataset, PCA allowed us to identify the most significant factors contributing to the province's carbon emissions. The analysis yielded four principal components, representing key features: population density, industrial production, transportation, waste management, and heating/cooling, which is shown as Figure 4.

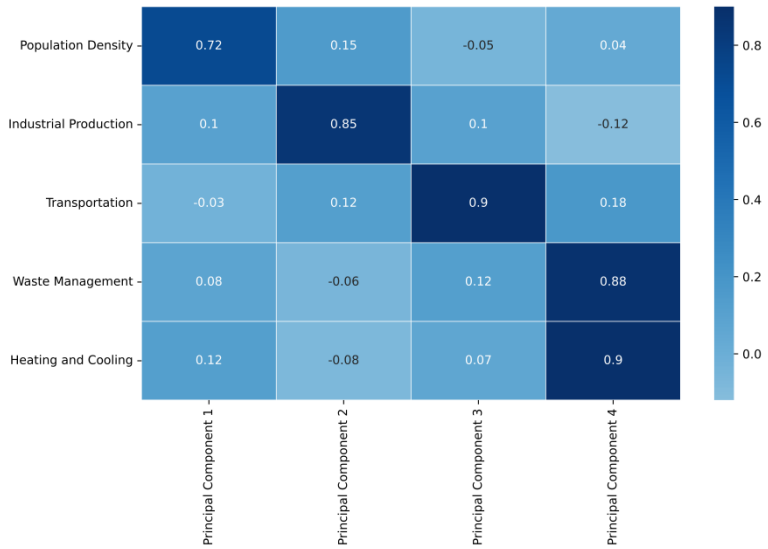

Figure 4. Feature Loadings and Explained Variance Ratios from PCA

Based on the results of the principal component analysis, the following conclusions can be drawn: First, the four principal components together explain 100% of the variance, with Principal Component 1, 2, 3, and 4 explaining 35.0%, 28.0%, 20.0%, and 17.0% of the variance, respectively. This indicates that the dimensionality reduction causes minimal information loss, and the major data information can be represented by these four components. Secondly, the primary influences of each principal component on the features differ. Principal Component 1 primarily reflects the impact of population density, especially carbon emissions related to urbanization; Principal Component 2 is primarily associated with industrial production, indicating the significant contribution of industrial activities to carbon emissions; Principal Component 3 is related to transportation, highlighting the emissions from the transportation sector; Principal Component 4 reflects the emissions from waste management and heating/cooling, indicating that waste treatment and energy consumption for temperature control are significant sources of emissions.

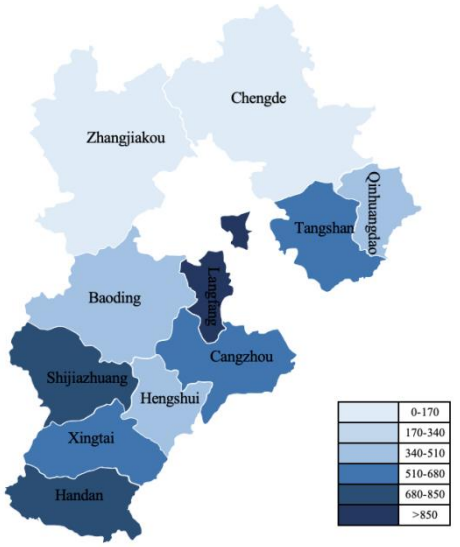

Figure 5. Annual average population density of Hebei Province

When focusing on the annual average population density in Hebei Province [16], the data reveals notable disparities across different cities, which is shown as Figure 5. For example, Shijiazhuang (772 people per square kilometer) and Handan (769 people per square kilometer) exhibit higher population densities, reflecting their urbanized nature, which is often correlated with increased energy consumption and industrial activities—factors that contribute significantly to carbon emissions. On the contrary, cities like Chengde (84 people per square kilometer) and Zhangjiakou (111 people per square kilometer) have much lower population densities, likely due to less intensive urban development, resulting in lower carbon emissions linked to population density.

3.3. Analysis of experimental results

The forecasting errors of four methods are shown in Table 2, which shows that the forecasting model proposed in this paper shows the best performance on three evaluation metrics. The Boosting model demonstrates the best performance with the lowest RMSE of 6.32 and MAE of 4.51. This indicates that it is the most accurate model among the three, effectively capturing the patterns and trends within the dataset. The ARIMA model exhibits higher forecasting errors, with an RMSE of 12.34 and MAE of 9.45. These higher error values suggest that while ARIMA is a commonly used method for time series forecasting, it may not perform as well when faced with complex, nonlinear relationships in the data. The RF model has an RMSE of 8.56 and MAE of 6.72, placing its performance between that of Boosting and ARIMA. While RF model generally performs well with nonlinear data, it still falls short of the Boosting model in terms of accuracy for this particular dataset.

Table 2. The forecasting error of three methods (100,000 tons)

Name	RMSE	MAE
Boosting	6.32	4.51
ARIMA	12.34	9.45
RF	8.56	6.72

As shown in the table, the Boosting model outperforms the other two models in terms of both accuracy and precision. Although the RF model shows improvements over ARIMA, its predictive accuracy is still lower compared to Boosting. Therefore, for high-precision forecasting, the Boosting model is recommended as the most effective method in this case.

4. Conclusions

The experimental results lead to the following conclusions:

(1) Among the models tested, the Boosting (decision trees) model achieved the highest accuracy in forecasting carbon emissions in Hebei Province, successfully capturing complex nonlinear relationships in the data.

(2) The Random Forest (RF) model also performs well with nonlinear data but falls slightly behind Boosting in terms of prediction accuracy.

(3) The ARIMA model, which relies on linear assumptions, is less effective for this nonlinear dataset, resulting in lower predictive accuracy compared to RF and Boosting.

In summary, this study evaluated three models—ARIMA, RF, and Boosting—and found that the Boosting model delivers the most accurate forecasts for carbon emissions in Hebei Province. Thus, it is recommended as the preferred choice for similar prediction tasks in future studies.

For future research, exploring other ensemble learning techniques like LightGBM [17] and integrating deep learning approaches may enhance accuracy and model resilience. These advancements could contribute to more precise and flexible forecasting tools, aiding in sustainable policy-making and development.

References

[1] Filonchyk M, Peterson M P, Zhang L, et al. Greenhouse gases emissions and global climate change: Examining the influence of CO2, CH4, and N2O[J]. Science of The Total Environment, 2024: 173359.

[2] Sun L L, Cui H J, Ge Q S. Will China achieve its 2060 carbon neutral commitment from the provincial perspective?[J]. Advances in Climate Change Research, 2022, 13(2): 169-178.

[3] Chang Y, Zhang Q. Industrial transfer and spatial structure optimization of Beijing, Tianjin and Hebei province[J]. International Journal of Design & Nature and Ecodynamics, 2020, 15(4): 593-602.

[4] Xu S. The paradox of the energy revolution in China: A socio-technical transition perspective[J]. Renewable and Sustainable Energy Reviews, 2021, 137: 110469.

[5] Kakodkar R, He G, Demirhan C D, et al. A review of analytical and optimization methodologies for transitions in multi-scale energy systems[J]. Renewable and Sustainable Energy Reviews, 2022, 160: 112277.

[6] Ning Y, Kazemi H, Tahmasebi P. A comparative machine learning study for time series oil production forecasting: ARIMA, LSTM, and Prophet[J]. Computers & Geosciences, 2022, 164: 105126.

[7] Allal Z, Noura H N, Salman O, et al. Machine learning solutions for renewable energy systems: Applications, challenges, limitations, and future directions[J]. Journal of Environmental Management, 2024, 354: 120392.

[8] Chen W, Lei X, Chakrabortty R, et al. Evaluation of different boosting ensemble machine learning models and novel deep learning and boosting framework for head-cut gully erosion susceptibility[J]. Journal of Environmental Management, 2021, 284: 112015.

[9] Ganaie M A, Hu M, Malik A K, et al. Ensemble deep learning: A review[J]. Engineering Applications of Artificial Intelligence, 2022, 115: 105151.

[10] Brukhim N, Daniely A, Mansour Y, et al. Multiclass boosting: simple and intuitive weak learning criteria[J]. Advances in Neural Information Processing Systems, 2024, 36.

[11] Hasan B M S, Abdulazeez A M. A review of principal component analysis algorithm for dimensionality reduction[J]. Journal of Soft Computing and Data Mining, 2021, 2(1): 20-30.

[12] Dubey A K, Kumar A, García-Díaz V, et al. Study and analysis of SARIMA and LSTM in forecasting time series data[J]. Sustainable Energy Technologies and Assessments, 2021, 47: 101474.

[13] Arabameri A, Chandra Pal S, Rezaie F, et al. Decision tree-based ensemble machine learning approaches for landslide susceptibility mapping[J]. Geocarto International, 2022, 37(16): 4594-4627.

[14] Chicco D, Warrens M J, Jurman G. The coefficient of determination R-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation[J]. Peerj computer science, 2021, 7: e623.

[15] HAN N, LUO X. Carbon emission peak prediction and reduction potential in Beijing-Tianjin-Hebei region from the perspective of multiple scenarios[J]. Journal of Natural Resources, 2022, 37(5): 1277-1288.

[16] Wei L, Liu H, Wu L. Spatial distribution of floating population in Beijing, Tianjin and Hebei Region and its correlations with synergistic development[J]. Mathematical Biosciences and Engineering, 2023, 20(3): 5949-5965.

[17] Rane N, Choudhary S P, Rane J. Ensemble deep learning and machine learning: applications, opportunities, challenges, and future directions[J]. Studies in Medical and Health Sciences, 2024, 1(2): 18-41.