Transportation CO_2 Emission Forecasting Method Based On Ensemble Learning Approach

Jagmeet Singh[1], Dr. Gurpreet Singh[2], Dr. Dapinder Deep Singh[3]

[1]Research Scholar, Civil Engineering, Shaheed Bhagat Singh State University, Ferozepur, Punjab, India.
[2]Assistant Professor, Civil Engineering, Shaheed Bhagat Singh State University, Ferozepur, Punjab, India.
[3]Assistant Professor, Civil Engineering, Shaheed Bhagat Singh State University, Ferozepur, Punjab, India.

Abstract

In this study, a transportation CO_2 emission forecasting method is developed using the ensemble learning approach. Initially, the dataset of transportation carbon emissions was collected from the open-source database. Following that, preprocessing and data splitting were done to train and validate the performance of the machine learning (ML) algorithms. In this work, six ML algorithms were individually trained and tested for the pre-processed data. Out of these algorithms, the best two algorithms were selected for designing the ensemble learning approach. In addition, the hyper-parameter tuning of the selected ML algorithms was done using the Optuna optimizer. The simulation result indicates that the proposed method achieves lower error metric values, such as an RMSE value of 38.536, an MAE value of 26.087, and a MAPE value of 0.014, and outperforms the existing approach.

Keywords: Carbon Emission, Ensemble Learning, Forecasting, Machine Learning, Optimization, Optuna.

1. Introduction

Global warming caused by humans is a severe and critical hazard with significant impacts on the environment and societies [1]. These adverse impacts, like rising sea levels, severe weather, lower crop yields, water shortages, and changes to ecosystems, are caused by greenhouse gases like nitrous oxide (N2O), carbon dioxide (CO_2), methane (CH4), ozone (O3), and chlorofluorocarbon (CFC). As a result, there are serious threats to people's health. Although many greenhouse gases are produced, CO_2 makes up about 81% of all emissions. Out of these, the transportation industry is currently responsible for 21% of carbon emissions worldwide. The International Energy Agency (IEA) predicts a 60% increase in vehicle use and a threefold boost in freight aviation by 2070 [2]. About 75% of all transportation is done by car, which is responsible for 15% of all carbon emissions globally. Passenger vehicles comprise 45.1% of these emissions, while trucks that carry products comprise 29.4%. Transportation systems like planes, trains, ships, and others comprise 11.6%, 10.6%, 1%, and 2.2%, respectively [3].

The transportation industry is the second greatest CO_2 emitter due to its size and rising energy use. In the European Union (EU), transportation is responsible for over 25% greenhouse gas emissions and around 25% of global energy consumption. The transportation sector's proportion of CO_2 emissions in EU nations rose from 32% to 45% between 1990 and 2015. To address the rising use of fossil fuels and transportation-related CO_2 emissions, a sustainable transportation sector and its reduction are crucial. This is especially significant for the 196 countries that signed the Paris Agreement on December 12, 2015 [2]. Therefore, in this paper, we have focused on an automatic forecasting method for transportation sector-related CO_2 emissions.

This article has a total of six sections. Section 1 defines the background and motivation. Section 2 presents the related work that reflects the previous approaches designed in this field for estimating the CO_2. chapter 3 defines the preliminaries required to design the method. Section 4 explains the proposed methodology is designed for CO_2 emission using the machine learning algorithms. Section 5 presents the results and analysis. Finally, Section 6 defines the conclusion and future scope of the work.

2. Related Work

Li et al. [2] used three ML algorithms, namely SVM, GBR, and OLS, to predict the transportation-based CO_2 emissions. The study used both transportation and socioeconomic factors in this study. The research studied the top 30 countries that produce CO_2 emissions. It included Tier 1 and Tier 2 groups. In Tier 1, the top 5 countries that are responsible for 60% for worldwide emissions were included. In Tier 2, the next 25 countries that are responsible for 35% of total emissions were included. The researcher evaluated the proposed model using 4-fold CV and calculated four performance metrics named RMSE, MAE, MAPE, and R^2. From the evaluated models, the GBR model performed well with an RMSE value of 0.1165, an MAPE value of 0.1408, and an R^2 value of 0.9943. The research also finds the significant socioeconomic and transportation features required to predict CO2 emissions. Sahreai et al. [4], aimed to develop a flexible, accurate, and open forecasting framework for US transportation-related CO_2 emissions. Therefore, opposed to traditional black-box models, they suggested a multi-stage approach that combines explainable

Machine Learning (ML) with Feature Selection (FS), ensuring interpretability. The application of hierarchical feature clustering and multi-step FS assisted in reducing the high multicollinearity among the 24 initial variables. As a result, five key predictors were identified: Total Fossil Fuels Consumption (FFT), Total Primary Energy Imports (TPEI), Annual Vehicle Miles Travelled (AVMT), Unemployment Rate (UR), and Air Passengers: Domestic and International (APDI). Four ML techniques were used: Multilayer Perceptron, eXtreme Gradient Boosting, ElasticNet, and SVR. ElasticNet performed better than the other three, with MAE = 30.6, RMSE = 45.53, and MAPE = 0.016. According to SHAP research, AVMT, FFT, and APDI are the main causes of CO_2 emissions. This methodology helped policymakers identify clear investment goals and make informed decisions. W. Wang and J. Wang [5], studied that one of the biggest sources of carbon emissions is known to be the transportation industry. To meet China's Paris Agreement carbon peak obligation on time, it is critical to explore peak carbon emissions and solutions for mitigation in the transportation sector. Few studies have forecasted China's transport CO_2 emissions peak, especially using an intelligent prediction algorithm.
This paper proposed a bio-inspired prediction model, the extreme learning machine (ELM) optimized by manta rays' foraging optimization (MRFO), to examine the determinants and effectively predict China's transport CO_2 emissions peak. The mean impact value (MIV) approach is then used to assess and distinguish the relative significance of thirteen contributing elements while adhering to this hybrid model. Furthermore, China's transportation-related CO_2 emissions are predicted using three different scenarios. According to the empirical findings, the suggested MRFO-ELM performs very well in terms of prediction accuracy and optimization aiming velocity. At the same time, one of the new significant elements influencing China's transportation-related CO_2 emissions is confirmed to be the degree of vehicle electrification. In China, the highest amount of CO_2 emissions from transportation would happen in 2039 under the base case model. In the scenarios for sustainable development and high growth, the emissions would level off in 2035 and 2043, respectively. Current pressure on China's transport carbon emissions abatement is due to peak years, however active policy modifications can accelerate emission peak. These new findings show that China must enhance its energy mix and push electric energy substitution in step with urbanization to reduce transport industry CO_2 emissions. Çınarer et al. [6], estimated CO_2 emissions in Turkey's transportation industry using three distinct AI algorithms: Extreme Gradient Boosting (XGBoost), Support Vector Machine (SVM), and Multi-layer Perceptron (MLP). The input parameters used are vehicle kilometers (VK), year (Y), energy consumption (ENERGY), population (POP), and GDP per capita. There are significant relationships found, with ENERGY showing the strongest association, followed by GDP, VK, POP, and Y. The correlation effect is used to make four scenarios: scenario 1 (ENERGY/VK/POP/Y/GDP), scenario 2 (ENERGY/VK/POP/Y), scenario 3 (ENERGY/VK/POP), and scenario 4 (ENERGY/VK). Experiments utilize statistical parameters (R2, RMSE, MSE, and MAE) to compare their impact on CO_2 emissions.

The range of R2 and RMSE values for all techniques and situations is 0.8969 to 0.9886 and 0.0333 to 0.1007, respectively. In case 4, the XGBoost algorithm works best. Li et al. [7], studied that urban areas are the primary source of greenhouse gas (GHG) emissions globally. As a result, they are becoming increasingly significant in mitigation efforts. Estimating CO_2 emissions at the city level is challenging because energy-related statistics data is often missing or of poor quality, especially in places that aren't well developed. This research looked at this problem and used open access data and ML to forecast and estimate CO_2 output at the city level across China. For CO_2 emission modeling, Recursive Feature Elimination and Boruta were employed to extract crucial variables and input values. Finally, 18 of 31 predictor factors were chosen for CO_2 emission prediction models. This study showed that industrial SO2 and dust released per capita are the most critical factors for forecasting city-level CO_2 emissions in China. Compared to other approaches, the XGBoost models had the lowest relative error (about 0.8%) and the best estimation accuracy (R2 > 0.98). The forecast accuracy of CO_2 emissions can be modestly enhanced by combining geographical and meteorological interpolation predictor factors. When they kept the other factors same, they saw that the link between urban CO_2 emissions per person and urban economic development was S-shaped instead of U-shaped. This study shows that ML algorithms can reliably forecast city-level CO_2 emissions using socioeconomic statistics records and geospatial data from metropolitan regions. For developing locations with little energy-related statistics data, this method can create carbon footprint maps often to help policymakers build particular carbon emissions reduction policies.

3. Preliminaries

- **Ridge Regression:** This type of regression is based on ordinary least squares (OLS). In this, the linear function is fitted using several explanatory parameters [8]. Let's assume we have Y as an explained variable, n data points, and X as explanatory parameters. Using this, the OLS model can be described as follows:

$Y_{(1\times n)=}X_{(n\times p+1)}\beta_{(p+1\times 1)} + \epsilon_{(1\times n)}$ (4.1)

In the above equation, ε describes the model error. Here, βs are calculated by reducing the error function as given below:

$\frac{1}{n}\sum_{i=1}^{n}(X\beta - y)^2$ (4.2)

But in the case of ridge regression, βs are calculated by reducing the penalized error. This is represented as follows:

$\frac{1}{n}\sum_{i=1}^{n}(X\beta - y)^2 + \alpha\sum_{j=0}^{p}\beta_j^2$ (4.3)

In the above equation, α > 0. This represents a complexity parameter that controls the amount of shrinkage. As a result, the ridge regression shrinks β to 0. This algorithm is highly significant when there exists a strong correlation between the explanatory parameters. This is also referred to as multicollinearity.

- **Lasso Regression:** The Least Absolute Shrinkage and Selection Operator (LASSO) is a type of linear regression. This algorithm includes regularization to select related features and prevent overfitting [9]. LASSO is different from traditional linear regression because it limits the absolute size of the regression coefficients. This makes the model sparser by reducing the size of less important coefficients to zero. This feature of the LASSO algorithm makes it suitable for model interpretation and feature selection. This is particularly important for high-dimensional datasets that contain feature redundancy.
- **ElasticNet:** This model includes L1 and L2 regularization penalties and thereby combines the strengths of both ridge and Lasso regression [9]. Since our goal is to examine the effects of regularization, this model wasn't chosen as a primary model. Ridge regression (with L2 regularization) was selected to compare it with other models that use the same regularization (L2). This helps to isolate and analyze the effects of different regularization strategies more clearly. With this hybrid technique, ElasticNet can manage feature selection and multicollinearity like the lasso algorithm. At the same time, it also offers improved accuracy of prediction and stability, like the ridge algorithm. Thus, by tuning the combined parameters, the ElasticNet model provides flexibility and a trade-off between model complexity and sparsity.
- **XGBoost:** XGBoost (Extreme Gradient Boosting) is a scalable and fast variant of gradient boosting. This algorithm builds a sequential ensemble of regression trees [10]. This algorithm includes a 2nd order Taylor approximation related to the loss function for accurate optimization. It also includes regularization to reduce overfitting and control regularization. Parallel computing is supported by XGBoost, and thus, it results in higher efficiency. XGBoost also supports optimized tree pruning and column subsampling. These make it a more effective approach for time-series prediction, like forecasting the CO2 emissions.

The model's hyperparameters include:

1. Learning rate (η) – It is the rate at which the tree's input affects the final prediction.
2. Number of estimators — It is the total number of ensemble trees.
3. Maximum tree depth — It describes the tree complexity.

- **Support Vector Regression (SVR):** This algorithm is related to statistical learning theory that uses a kernel function to transform input data into a higher-dimensional space. The objective here is to identify the optimal hyperplane, which balances accuracy and complexity and maximizes the margin of tolerance [10]. The SVR model is described using kernel function K (x_i, x), Lagrange multipliers (α_i), and bias term b (eqn. 10). The algorithm also utilizes linear and RBF kernels to find the linear and non-linear patterns in CO2 emissions. Furthermore, the algorithm is optimized by tuning epsilon (ε) for tolerance, kernel, and C (regularization parameter). This balances the model's training error and simplicity, thus preventing overfitting.

$f(x) = \sum_{i=1}^{n} a_i K(x_i, x) + b$ (4.4)

In the above Equation, $f(x)$, $a_i K, x_i$, b denotes the predicted output, support vector, Lagrange multiplier estimated during training, kernel function, and bias term.

4.2.6 KNN Regressor

This is a supervised learning method that is simple and effective for both regression and classification problems [11]. This algorithm works by predicting the value or class of data by considering its neighbour value in the training set. The basic principle of KNN is "like is close to like." With a new data point, the algorithm estimates the distance between all data points in the training set and that new point. The algorithm then identifies the k-neighbours that have a small distance and assigns classes. These classes are based on most of the neighbours and are used to perform prediction using regression based on average values. This algorithm is easy to implement and offers effective results for medium and small-sized datasets. On large datasets, the algorithm's computational costs can be higher. Also, the selection of distance metrics might also affect performance. The equation below shows the mathematical expression of KNN:

$\hat{y}(x^*) = \frac{1}{k} \sum_{i \epsilon N_k(x^*)} y_i$ (4.5)

4.2.7 Gradient Boosting

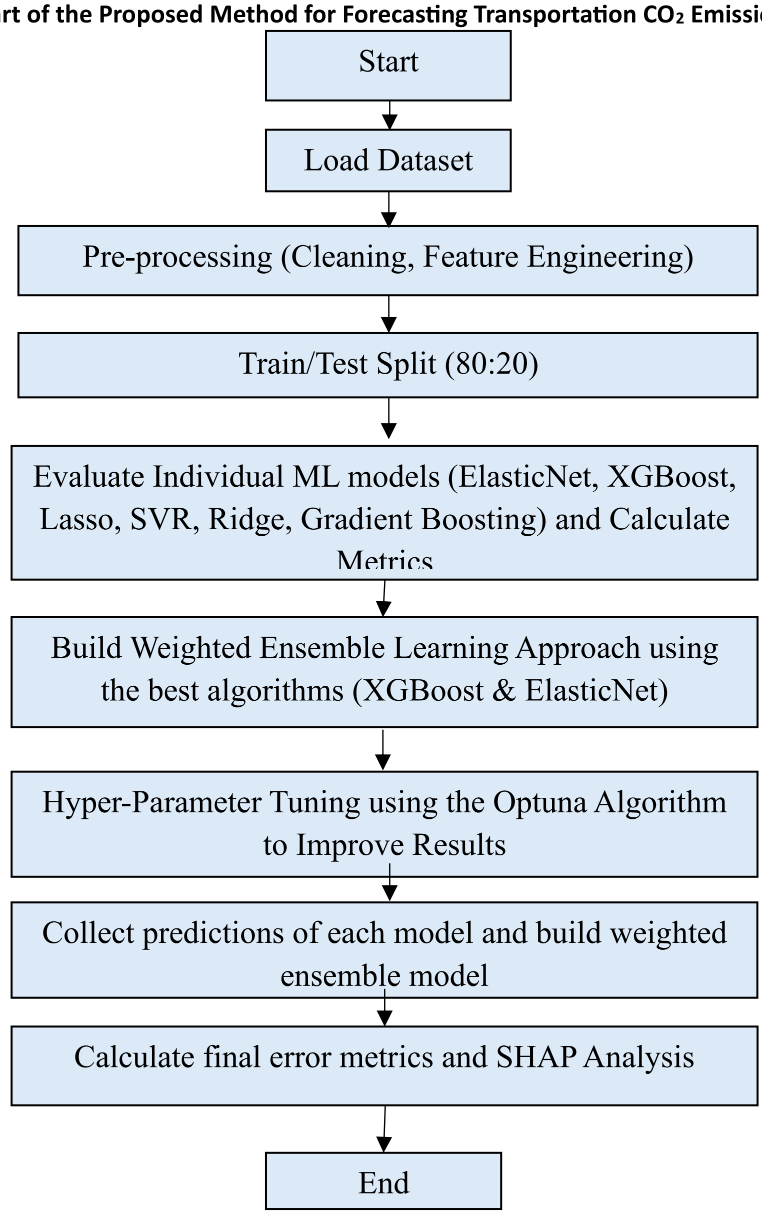
This algorithm works by creating regression trees. In this regression tree, each new tree is created to identify and correct the preceding node's residual errors [10]. The process of this algorithm starts with a basic regression tree, and later an iterative step of tree-building that divides the dataset into small sets. Each new tree is trained in a way to reduce the errors of the previous steps. This process is repeated for a specified number of trees or until the model is fitted. Each tree has a learning rate to regulate the tree's contribution and mitigate overfitting. This further improves the generalization. To improve the GB model's performance, key parameters include the number of estimators, the depth of trees, and the learning rate. These parameters balance the robustness and accuracy.

4. Proposed Transportation CO_2 Emission Forecasting Method

This section presents the proposed transportation CO_2 emission forecasting method based on the ensemble learning approach. The main novelty of the designed method is that the best two ML algorithms were chosen out of six ML algorithms to design the ensemble learning approach, and hyperparameter tuning of the selected ML algorithms was done using the Optuna optimiser. Figure 1 shows the flowchart of the proposed CO_2 emission forecasting transportation method and its detailed description is given below.

The proposed method focuses on predicting CO_2 emissions using several related variables like economic, energy, and transport-related variables. In the proposed methodology, first, the dataset is loaded and preprocessed to handle missing values. After that, a few abnormal points from the dataset are removed, like the year 2020, to avoid abnormal trends. To add temporal features, lag features of TCT are developed using the previous time stamps (t-1 to t-4). After adding these features, rows with null values are removed. The preprocessed data is then divided into a training and testing ratio of 80:20. The proposed methodology uses several ML models, including SVR, ElasticNet, Ridge, XGBoost, Lasso, GB, and KNN. The performance of these algorithms is evaluated using RMSE, MAE, and MAPE metrics. After analyzing the performance of individual models, a hybrid model is developed that combines XGBoost and ElasticNet using a weighted average approach. The hybrid model uses optimized parameters (that were optimized using OPTUNA to minimize RMSE). After that, the hybrid model is retained with the best parameters. In the end, the performance metrics are compared to baseline models.

Figure 1. Flowchart of the Proposed Method for Forecasting Transportation CO_2 Emission



5. Results and Analysis

This section presents the evaluation of the proposed method for forecasting transportation CO_2 emissions using the ensemble learning approach and comparative analysis with the existing approaches. The Python language was used to write the code, and Google Colab software was used for simulation purposes.

5.1 Parameter Configuration for Simulation Evaluation

In the evaluation criteria, the initialization process of the algorithms that are utilized in the proposed method is defined. Table 1 summarized the parameter values that were taken for evaluation purposes.

Table 1. Algorithms Parameter Value Utilized in the Proposed Method for Evaluation Purposes

Parameter	**Value**
Sample Record	1990-2023
Total Number of Attributes	25
Training and Testing Ratio	80:20
Parameter Value of ML Algorithms	
ElasticNet	alpha = 0.001 l1_ratio = 0.9
SVR	kernel = RBF C = 10 gamma = scale epsilon = 0.1
XGBoost	n_estimators = 200 max_depth = 3 learning_rate = 0.05 subsample = 0.9 colsample_bytree = 0.9
Ridge	alpha = 1.0
Lasso	alpha = 0.01
KNN	n_neighbors = 5
Gradient Boosting	n_estimators = 200 learning_rate = 0.05 max_depth = 3
Hybrid Model	ElasticNet + XGBoost
Weight Value of the Voting Classifier	[0.4,0.6]
Optuna Algorithm	Bayesian Hyperparameter Optimization using Tree-structured Parzen Estimator (TPE)
Number of Optuna Trials	100
Cross Validation Method	TimeSeriesSplit (5 splits)
Total Execution time	117.2869 seconds

5.2 Performance Metrics

this section defines the performance indices used to evaluate the proposed method's performance. Furthermore, these indices are also used for comparative analysis purposes to validate the performance of the proposed method.

Table 2. Performance Indices

Parameters	**Description**	**Equation**

RMSE	This parameter measures the root mean square error between the actual and predicted values. This parameter is maximally utilized in the carbon emission method due to the large deviation between values.	$RMSE = \sqrt{\sum_{i=1}^{n} \frac{(y_i - \hat{y}_i)^2}{n}}$ (1)
MAE	This parameter determines the average absolute value between the actual and predicted.	$MAE = \sum_{i=1}^{n} \frac{\lvert y_i - \hat{y}_i \rvert}{n}$ (2)
MAPE	It expresses the prediction error as a percentage of the actual value.	$MAPE = \frac{1}{n}\sum_{i=1}^{n} \left\lvert \frac{y_i - \hat{y}_i}{y_i} \right\rvert \times 100$ (3)

In Eqs. (1-3), n denotes the total number of samples, $y_i, \hat{y}_i$ denotes the actual and predicted value.

5.3 Simulation Results

Table 3 shows the evaluation of the individual ML algorithms utilized in the proposed method. This evaluation was performed by splitting the dataset into an 80:20 ratio of training and testing. The result indicates that the ElasticNet algorithm achieves the highest value of error metrics, whereas XGBoost achieves the lowest due to handling the non-linearity between the attributes of the dataset more effectively than other algorithms.

Table 3. Evaluation of the Individual ML Algorithms Utilized in the Proposed Method

ML Algorithm	RMSE	MAE	MAPE
ElasticNet	**66.032**	**42.847**	**0.023**
SVR	50.728	41.564	0.022
XGBoost	**43.211**	**34.190**	**0.018**
Ridge	58.929	38.165	0.021
Lasso	65.883	42.740	0.023
KNN	54.047	48.833	0.026
GB	61.113	38.853	0.021

From the above results, it is evident that XGBoost performed extremely well as compared to other selected models in terms of RMSE, MAE, and MAPE metrics. In the previous studies [1], the ElasticNet model performed superior to other models. So, in this work, an ensemble learning approach was designed using XGBoost and ElasticNet to improve the prediction performance. After that, a weighted voting classifier was used for the final prediction. Besides that, the hyper-tuning of the parameters of selected ML algorithms was done using the Optuna optimizer. The result indicates that the proposed method achieves the RMSE value of 38.5360, MAE value of 26.0870, and MAPE value of 0.0142 due to tuning the XGBoost and ElasticNet algorithms' parameters with the best values in the lower and upper limits of the parameters, as shown in Table 4. These error parameter values are lower than the previous ML algorithms and ensemble learning approach without hyper-tuning.

Table 4. Evaluation of the Individual ML Algorithms Utilized in the Proposed Method

ML Algorithm	RMSE	MAE	MAPE
ElasticNet	66.032	42.847	0.023
SVR	50.728	41.564	0.022
XGBoost	43.211	34.190	0.018
Ridge	58.929	38.165	0.021
Lasso	65.883	42.740	0.023
KNN	54.047	48.833	0.026
GB	61.113	38.853	0.021
Proposed Method	**38.5360**	**26.0870**	**0.0142**

5.4 Comparative Analysis

Table 5 presents the comparative analysis of the proposed method for forecasting transportation CO_2 emissions with respect to the existing recent method designed by Sahraei et al. [4] based on three performance indices, namely, RMSE, MAE, and MAPE. The result indicates that the proposed method improves significantly the RMSE value by 15.36%, the MAE value by 14.75%, and the MAPE value by 12.50% over the existing method proposed by Sahraei et al. [4] due to the ensemble learning approach and hyperparameter tuning using the Optuna algorithm.

Table 5. Comparative Analysis

Parameter	Sahraei et al. [4]	Proposed Method
RMSE	45.530	38.536
MAE	30.600	26.087
MAPE	0.016	0.014

6. Conclusion

The proposed method effectively forecasts the transportation carbon emission by processing the non-linear relationship between the dataset attributes due to utilizing machine learning algorithms. Furthermore, the ensemble learning approach and hyperparameter tuning using the Optuna optimizer help select the best ML algorithms and tune the parameters to their best value. The comparative analysis indicates that the proposed method significantly reduces the RMSE, MAE, and MAPE values over the existing method [4]. The implication of the proposed method is that it handles complex data effectively, helps in proactive policy and infrastructure planning, and provides precision in micro-level forecasting. The limitation of the presented method is that the data sample is limited and time complexity is high due to employing several ML algorithms and hyperparameter tuning the parameters using the Optuna optimizer.

References

[1] A. A. Ajala, O. L. Adeoye, O. M. Salami, and A. Y. Jimoh, "An examination of daily CO2 emissions prediction through a comparative analysis of machine learning, deep learning, and statistical models," Environmental Science and Pollution Research, vol. 32, no. 5, pp. 2510–2535, Jan. 2025, doi: 10.1007/s11356-024-35764-8. Available: https://doi.org/10.1007/s11356-024-35764-8

[2] X. Li, A. Ren, and Q. Li, "Exploring patterns of Transportation-Related CO2 emissions using machine learning methods," Sustainability, vol. 14, no. 8, p. 4588, Apr. 2022, doi: 10.3390/su14084588. Available: https://doi.org/10.3390/su14084588

[3] D. L. Albritton et al., Climate Change 2001: Synthesis Report. Cambridge University Press, 2001. Available: http://www.cambridge.org

[4] M. A. Sahraei, K. Li, and Q. Qiao, "A Multi-Stage feature selection and explainable machine learning framework for forecasting transportation CO2 emissions," Energies, vol. 18, no. 15, p. 4184, Aug. 2025, doi: 10.3390/en18154184. Available: https://doi.org/10.3390/en18154184

[5] W. Wang and J. Wang, "Determinants investigation and peak prediction of CO2 emissions in China's transport sector utilizing bio-inspired extreme learning machine," Environmental Science and Pollution Research, vol. 28, no. 39, pp. 55535–55553, Jun. 2021, doi: 10.1007/s11356-021-14852-z. Available: https://doi.org/10.1007/s11356-021-14852-z

[6] G. Çınarer, M.K. Yeşilyurt, U. Ağbulut, Z. Yılbaşı, and K. Kılıç, "Application of various machine learning algorithms in view of predicting the CO2 emissions in the transportation sector," Science and Technology for Energy Transition, vol. 79, no. 15, 2024.

[7] Y. Li and Y. Sun, "Modeling and predicting city-level CO2 emissions using open access data and machine learning," Environmental Science and Pollution Research, vol. 28, no. 15, pp. 19260–19271, Jan. 2021, doi: 10.1007/s11356-020-12294-7. Available: https://doi.org/10.1007/s11356-020-12294-7

[8] T. Mahmood and M. Asif, "Prediction of energy efficiency for residential buildings using supervised machine learning algorithms," Energies, vol. 17, no. 19, p. 4965, Oct. 2024, doi: 10.3390/en17194965. Available: https://doi.org/10.3390/en17194965

[9] Y. Dani, N. Belouaggadia, and M. Jammoukh, "Predicting CO2 emissions in Morocco: exploring the use of ridge regression with data preprocessing and feature impact analysis," Environmental Science and Pollution Research, vol. 32, no. 45, pp. 25618–25642, Nov. 2025, doi: 10.1007/s11356-025-37156-y. Available: https://doi.org/10.1007/s11356-025-37156-y

[10] M.P. Keykele and M. Ravanshadnia, "Predicting residential CO2 emissions: Machine learning and deep learning approaches for short-term forecasting," International Journal of Sustainable Building Technology and Urban Development, vol. 16, no. 3, pp. 373-387, 2025. Available: https://doi.org/10.22712/susb.20250024

[11] M. S. Mungan and O. Arpa, "Estimation of CO2 emissions from vehicles using machine learning and multi-model investigation," Bulletin of the Polish Academy of Sciences Technical Sciences, p. 154287, Mar. 2025, doi: 10.24425/bpasts.2025.154287. Available: https://doi.org/10.24425/bpasts.2025.154287