ADBA Computer Science
Cutting-Edge Scientific Solutions

e-ISSN: 3023-8544
RESEARCH ARTICLE
Vol.1 / No.1 / 2024 / pp.19-25
https://doi.org/10.69882/adba.cs.2024074

Environmental Sustainability through AI: A Case Study on CO_2 Emission Prediction

Cem Özkurt [*,α,1]

*Department of Computer Engineering, Sakarya University of Applied Sciences, 54050, Sakarya, Turkiye, αArtificial Intelligence and Data Science Application and Research Center, Sakarya University of Applied Sciences, 54050, Sakarya, Turkiye.

ABSTRACT In this study, the Biogeography-Based Optimization (BBO) algorithm was effectively utilized to predict carbon dioxide (CO_2) emissions. In the context of combating global warming and climate change, making accurate and reliable CO_2 emission predictions is critically important for developing environmental policies and strategies. Accordingly, the motivation for our study is to contribute to environmental decision-making processes by improving the accuracy of CO_2 emission predictions. BBO is a nature-inspired optimization method used to analyze complex relationships and identify significant features within a dataset. The focus of the study is to accurately predict the "share_global_coal_co2" parameter, and for this purpose, the BBO algorithm was employed to identify the 20 most influential features. The analyses revealed that the Gradient Boosting algorithm provided the lowest Mean Squared Error (MSE) value of 0.347408, indicating that the model can make predictions closer to the actual data. Additionally, the use of interpretable artificial intelligence models such as SHAP and LIME made the model's predictions more understandable and clearly demonstrated the impact of specific features on the predictions. The results obtained provide significant guidance for environmental policymakers and energy experts. The effectiveness of the BBO algorithm in predicting CO_2 emissions can contribute to more informed and data-driven decisions in environmental analysis and policy-making processes. This study emphasizes the importance of artificial intelligence and optimization techniques in achieving sustainability goals and helps develop more effective strategies in environmental management.

KEYWORDS
CO_2 emissions
Biogeography-Based optimization (BBO)
Explainable AI
SHAP
LIME

INTRODUCTION

Addressing global environmental issues and climate change, the reduction of carbon dioxide (CO_2) emissions and CO_2 capture technologies play a crucial role. In this context, the Biogeography-Based Optimization (BBO) algorithm has been employed to estimate the "share_global_coal_co2" parameter. Inspired by natural biogeographic processes, BBO is an optimization algorithm used to analyze complex relationships and identify significant features within a dataset. This study utilizes a dataset comprising various parameters, including share global coal CO_2, share global cumulative gas CO_2, share global cumulative LUC CO_2, share global cumulative flaring CO_2, cumulative oil CO_2, share global gas CO_2, share global cumulative coal CO_2, cumulative CO_2 including LUC, total GHG, cumulative LUC CO_2, share global cumulative cement CO_2, CO_2 growth percentage, cumulative cement CO_2, CO_2 including LUC per GDP, oil CO_2, coal CO_2, temperature change from GHG, consumption CO_2 per GDP, share global LUC CO_2, cement CO_2, cumulative coal CO_2, primary energy consumption, other industry CO_2, share global cumulative oil CO_2, CO_2 per GDP, cumulative gasCO_2, temperature change from CO_2, nitrous oxide, share global cumulative CO_2 including LUC, and share global CO_2.

By applying the BBO algorithm, the study aims to reduce these parameters to the 20 most influential ones for accurately predicting the "share_global_coal_co2" parameter. Artificial intelligence techniques significantly contribute to reducing CO_2 emissions and developing sustainable energy solutions. For instance, Delanoë *et al.* (2023) evaluated the positive and negative impacts of AI mod-

Manuscript received: 16 May 2024,
Revised: 13 June 2024,
Accepted: 25 June 2024.

[1]cemozkurt@subu.edu.tr (**Corresponding author**)

els on CO_2 emissions reduction. In this study, three different AI models were utilized for energy demand management in Brazilian households, photovoltaic power forecasting in Tunisia, and the electric vehicle routing problem in Sweden and Luxembourg. The results indicated that AI models can achieve significant CO_2 reductions depending on the context. Yan *et al.* (2020) developed a hybrid artificial intelligence model to predict the physical and chemical changes in coal seams during CO_2 geological sequestration. This model, integrating back propagation neural network (BPNN), genetic algorithm (GA), and adaptive boosting algorithm (AdaBoost), was optimized to accurately predict coal strength alterations due to CO_2 adsorption. The study demonstrated that the hybrid model could effectively and accurately predict these changes. Qerimi and Sergi (2022) examined the legislative processes related to carbon capture and storage (CCS) technology. This study emphasized the importance of CCS and AI technologies in achieving climate goals and argued for the necessity of new regulations to govern the development, design, and deployment of such technologies.

Another study explored the use of artificial neural network (ANN) tools to enhance the efficiency of CO_2 storage projects by predicting critical performance indicators like methane recovery and CO_2 injection. The findings showed that ANN models could accurately predict performance in CO_2 storage projects. Thanh *et al.* (2022) used hybrid artificial intelligence models to predict the deliverability of underground natural gas storage sites. This study highlighted the importance of developing intelligent systems that can accurately predict natural gas storage deliverability in various geological formations. Stef *et al.* (2023) investigated the impact of high-quality institutional measures on global CO_2 emissions reduction.

The study revealed that effective climate change policies must be associated with improvements in property rights protection, citizen participation in elections and freedom of expression, and corruption control. Heo *et al.* (2022) developed an explainable artificial intelligence (XAI) model to create a net-zero carbon roadmap for the petrochemical industry. This model produced various scenarios of offshore wind power and conducted techno-economic and environmental assessments. The findings underscored the feasibility and effectiveness of AI-driven net-zero carbon solutions.

This literature review highlights the significance and application areas of artificial intelligence and optimization algorithms in reducing CO_2 emissions. By examining the effectiveness of the BBO algorithm in predicting the "share_global_coal_co2" parameter, this study aims to contribute to more informed and data-driven decision-making processes in environmental analysis and policy development.

MATERIALS AND METHODOLOGY

This dataset contains various climate variables, greenhouse gas emissions, and economic indicators. Compiled to examine global carbon emissions and the environmental impact of various human activities, this dataset consists of a total of 32 different parameters. Parameters like "share_global_coal_co2" represent coal-related carbon dioxide emissions, while others like "cumulative_oil_co2" and "cumulative_gas_co2" indicate emissions from oil and gas sources, respectively. Economic indicators such as "gdp" and "consumption_co2_per_gdp" can be used to analyze the relationship between economic growth and greenhouse gas emissions. This dataset can be utilized with various machine learning models to predict the "share_global_coal_co2" parameter and can play a significant role in issues such as energy policy development and climate change strategy determination.

Dataset

This dataset contains various climate variables, greenhouse gas emissions, and economic indicators. Compiled to examine global carbon emissions and the environmental impact of various human activities, this dataset consists of a total of 32 different parameters. Parameters like "share_global_coal_co2" represent coal-related carbon dioxide emissions, while others like "cumulative_oil_co2" and "cumulative_gas_co2" indicate emissions from oil and gas sources, respectively. Economic indicators such as "gdp" and "consumption_co2_per_gdp" can be used to analyze the relationship between economic growth and greenhouse gas emissions. This dataset can be utilized with various machine learning models to predict the "share_global_coal_co2" parameter and can play a significant role in issues such as energy policy development and climate change strategy determination.

Biogeography-based Optimization (BBO)

BBO is a natural optimization algorithm based on biogeography principles, inspired by natural processes modeling the dispersion and migration of biological species among habitats. This algorithm utilizes mathematical models representing the quality of habitats and migration rates between species to optimize the fitness values of the population. The fundamental working principle of BBO involves the movement of the best individuals among habitats to improve the fitness values of a population initially generated randomly. This process enhances and optimizes the fitness values of the population over time. BBO can be effective in complex and multi-dimensional optimization problems, although it may require appropriate parameter settings and modeling tailored to the problem context.

Machine Learning

Machine learning is a branch of artificial intelligence where computer systems have the ability to learn from data. These systems create models using data to perform specific tasks or solve problems, and they can analyze new data or make predictions using these models. Machine learning is data-driven as it relies on learning from experiential data. Fundamentally, it is a combination of disciplines such as data analysis, statistics, mathematics, and computer science. Machine learning algorithms are commonly used in various tasks such as classification, regression, clustering, dimensionality reduction, and pattern recognition. Examples include decision trees, support vector machines, gradient boosting machines, and deep learning networks. Machine learning has a wide range of applications across various industries and fields, including healthcare, finance, automotive, retail, and more. However, training these models requires careful management of factors such as proper hyperparameter tuning and data quality.

Artificial Neural Networks (ANNs) are a machine learning model that mimics the workings of the human brain and has been successfully used in many fields in recent years. This model enables information processing and learning by forming a network of neurons, the basic units of a neural network. Artificial neural networks have structures consisting of multiple layers; each layer receives inputs from the previous layer and processes them. These processes are typically carried out with non-linear activation functions. Artificial neural networks can handle a wide range of data but may require large amounts of data and have lengthy training times. However, using a subfield called deep learning, they can exhibit superior performance in large and complex datasets. One of the fundamental advantages of artificial neural networks is their ability to optimize learning capabilities with various architectures and

hyperparameters. However, it's important to deal with issues such as overfitting and ensure good generalization to data outside the training set.

$$z = \sum_{i=1}^{n} w_i x_i + b \tag{1}$$

In Equation 1, "z" represents the value of the objective function, while "w_i" and "x_i" represent the components of the weight and input data, respectively. "b" is a constant term. The summation calculates the value of this objective function by taking a combination of weights and inputs over a specific dataset. Thus, the BBO algorithm attempts to find the best solution in a particular problem by optimizing this equation.

XGBoost Recently, XGBoost (eXtreme Gradient Boosting) has become increasingly popular for classification and regression problems, especially for structured data, yielding effective results. This machine learning algorithm constructs a strong predictor by combining many weak predictors, often referred to as decision trees. Using a technique called gradient boosting, at each step of the model, a new predictor is added to minimize the loss (error). XGBoost applies this gradient descent to decision trees and sequentially adds weak predictors. Thus, the model learns increasingly complex relationships and makes more accurate predictions (Chen and Guestrin 2016). One significant advantage of XGBoost is its ability to be optimized for different datasets and problems by adjusting its hyperparameters. Additionally, it can work quickly and handle large datasets well. However, its high performance and flexibility come at a cost. As the complexity of the model increases, both training and prediction times may increase. Therefore, it's essential to consider factors like computational resources and hyperparameter optimization when using XGBoost.

$$F(x) = L(\theta) + \Omega(\theta) \tag{2}$$

In Equation 2, "$F(x)$" represents the predicted value of the target variable, "$L(\theta)$" represents the loss function, and "$\Omega(\theta)$" represents the regularization term. Essentially, the XGBoost algorithm adds weak predictors sequentially by minimizing the residuals from the previous model's predictions. In this equation, "$L(\theta)$" calculates the error between the predicted and actual values of the features, and "$\Omega(\theta)$" is the regularization term that limits the complexity of the model. Thus, the XGBoost algorithm aims to optimize prediction performance by creating the most suitable model through the combination of the loss function and regularization term.

$$L(\theta) = \sum_{i=1}^{n} -(y_i \log\log(\hat{y}_i) + (1 - y_i) \log\log(\hat{y}_i)) \tag{3}$$

In Equation 3, the expression "$L(\theta)$" represents a loss function that measures how far the model's predictions are from the true labels. θ represents the model parameters. In the equation, "y_i" symbolizes the true label values, and "$\hat{y}_i$" represents the predictions made by the model. This loss function, used for binary classification problems, calculates the negative log probability sum over the predicted class probabilities for each data point's true class labels. Thus, the algorithm aims to learn the best model parameters by minimizing this loss function and aims to increase classification accuracy.

$$\Omega(\theta) = \gamma T + \frac{\lambda}{2} \sum_{j=1}^{T} w_j^2 \tag{4}$$

In Equation 4, the expression "$\Omega(\theta)$" represents a regularization term that limits the complexity of the model. γ and λ are regularization parameters and regularization coefficients, respectively. In the equation, "T" represents the number of trees, and "w_j" represents the weights of each tree. This regularization term is used to reduce the tendency of the model to overfit and improve its generalization ability. The first term controls the number of trees and their complexity, while the second term regulates the complexity of the trees by the square of their weights. This regularization term aims to prevent overfitting by helping the model become simpler and more generalizable. Thus, the XGBoost algorithm uses this regularization term to control the complexity of the model while minimizing the loss function.

LightGBM is a machine learning algorithm that has gained popularity recently, particularly for working effectively with large datasets. This algorithm constructs decision trees using the gradient boosting method, but unlike other traditional gradient boosting-based methods, LightGBM builds trees by considering specific features. This allows it to create more efficient trees by taking into account different levels of importance of features in the dataset (Zhang *et al.* 2020). LightGBM can handle large datasets well because it utilizes parallel computing capabilities to reduce training and prediction times. Additionally, it offers advantages such as low memory usage and high scalability. However, it's essential to properly adjust some hyperparameters of LightGBM; otherwise, you might encounter issues like overfitting or other problems that could negatively impact the model's performance.

$$F_m = F_{m-1}(x) + \eta \cdot h_m(x) \tag{5}$$

In Equation 5, "F_m" represents the m-th prediction of the function, while "$F_{m-1}(x)$" represents the sum of predictions made in the (m-1)th stage. η represents the learning rate, and "$h_m(x)$" represents the weak predictor added in the m-th stage. LightGBM constructs a tree-based model using the gradient boosting method. This equation shows adding a new tree to the current predictions at each stage of the model and adding the predictions of the tree with the learning rate to the total predictions. This process allows the model to learn complex relationships in the dataset without increasing complexity and preventing overfitting. Thus, LightGBM is successfully used across a wide range of applications, providing high accuracy and fast training times.

Random Forest (RF) is a widely used machine learning algorithm for classification and regression problems. This algorithm constructs a predictor by aggregating many decision trees. Each decision tree is trained on a subset of data, randomly sampled from the original dataset (bootstrap sampling), and features selected randomly (subspace sampling). Then, each tree makes its prediction, and in the case of classification, the final prediction is made by voting, or in regression, by taking the average. Random Forest can learn more complex decision boundaries than a single tree and reduces the risk of overfitting. Additionally, it is robust to noise and missing data in the input dataset. However, it's crucial to adjust RF's hyperparameters (e.g., number of trees, size of feature subsets, etc.), as otherwise, its performance may degrade or overfitting may ocur (Liaw and Wiener 2002).

$$f(x) = \frac{1}{N} \sum_{i=1}^{N} f_i(x) \tag{6}$$

In Equation 6, "$f(x)$" represents the predicted value of the function, "N" is the number of data points, and "$f_i(x)$" represents the

prediction of each decision tree. The Random Forest algorithm is an ensemble learning technique where many decision trees are collectively constructed, and the average prediction of each tree is taken. In this equation, "$f_i(x)$" shows the prediction made by each tree individually, while the term "$\frac{1}{N}$" provides the final prediction by averaging the predictions of all trees. This method balances the variance and errors within each tree while collectively obtaining a stronger and more generalized prediction. As a result, the Random Forest algorithm addresses complexity in the dataset while reducing the risk of overfitting, thus providing stable and accurate predictions.

Support Vector Machine (SVM) is a powerful machine learning algorithm used for classification and regression tasks. It determines a decision boundary in the feature space for classification or predicts a regression function, aiming to achieve the widest possible margin between classes, supported by a subset of training data points called support vectors (Gunn 1998). SVM works effectively on linearly separable problems and can handle nonlinear problems by transforming the feature space using kernel functions. Its advantages include effectiveness with high-dimensional datasets, reducing the risk of overfitting, and its ability to handle various data structures through different kernel functions. However, it's crucial to adjust SVM's hyperparameters (e.g., the C parameter, kernel type, etc.) correctly to avoid performance degradation or overfitting.

$$f(x) = w^T x + b \tag{7}$$

In Equation 7, "$f(x)$" represents the predicted classification for input data "x", while "w" denotes the weight vector and "b" stands for the bias term. SVM classifies data by creating a separation line between two classes. The "w" vector indicates the normal and slope of the separation plane, while the "b" bias term represents the distance of the plane from the origin. SVM utilizes this equation to find an optimal separation plane and is typically effective in classification problems.

$$margin = \frac{2}{\|w\|} \tag{8}$$

Equation 8 defines a concept known as the "margin" in the Support Vector Machine (SVM) algorithm. The "margin" indicates how far the separation plane is from the data points. The equation divides the norm (length) of the weight vector "$\|w\|$" and multiplies the result by two to calculate the margin. SVM aims to maximize the margin to find the best separation plane or hyperplane. Thus, SVM typically provides a wide margin for classifying data points and can better adapt to new data.

$$minimize \frac{1}{2}\|w\|^2 \tag{9}$$

Equation 9 represents the norm (length) of the weight vector "$\|w\|$". The goal of SVM is to classify data points by creating a separation plane between two classes. This equation is a mathematical expression used to determine the decision boundaries of SVM. By using this equation, SVM optimizes the weight vector and bias term (b) to find the equation of the separation plane (or hyperplane). Thus, it creates a separation plane that best classifies the data points. In summary, SVM aims to minimize the result of this equation to classify data points optimally.

$$K(x_i, x_j) = \phi(x_i)^T \phi(x_j) \tag{10}$$

Equation 10 represents the kernel function in the Support Vector Machine (SVM) algorithm. It denotes the measure of the dot product between two input vectors. "$\phi(x)$" symbolizes a feature map that transforms the input data into a higher-dimensional feature space. This function enables SVM to classify linearly inseparable data. By using this kernel function, SVM makes the data linearly separable and then finds a separation plane (or hyperplane). This kernel function plays a critical role in making the data linearly separable.

K-Nearest Neighbors (KNN) is a widely used machine learning algorithm for classification and regression problems. This algorithm relies on the nearest neighbors of a data point to determine its class or value. The distance between each data point and all other points in the feature space is calculated, and then the K nearest neighbors of the input data point are selected. In classification, a prediction is made based on the classes of these neighbors, while in regression, the average of the neighbors' values is used. KNN is a non-parametric algorithm, meaning it makes no assumptions about the underlying data distribution. It's also a versatile algorithm that can effectively handle both numerical and categorical data. However, it may suffer from computational inefficiency with large datasets, and selecting the correct value for K is crucial as it can affect the performance of the model (Kramer and Kramer 2013).

$$d(x_i, x_j) = \sqrt{\sum_{p=1}^{n} (x_{i,p} - x_{j,p})^2} \tag{11}$$

In Equation 11, "$d(x_i, x_j)$" represents the Euclidean distance between two data points, "x_i" and "x_j". "n" denotes the dimensionality of the data points, while "$x_{i,p}$" and "$x_{j,p}$" respectively represent the "p"-dimensional features of data points "i" and "j". The KNN algorithm uses the labels of neighboring data points to classify a new data point. This distance measurement determines the similarity or distance between one data point and another. The KNN algorithm classifies a new data point by considering the closest neighbors up to a specified "k" number. This distance measurement plays a fundamental role in the classification process of KNN, evaluating the relationship between data points to provide the most appropriate classification.

$$y = \text{mode}(y_1, y_2, \ldots, y_k) \tag{12}$$

In Equation 12, "y" represents the predicted value of the target variable, while "$y_1, y_2, \ldots, y_k$" denote the class labels of neighboring data points. The "mode" function determines the most frequently occurring class label among the neighboring data points, i.e., it takes the mode value. The KNN algorithm considers the labels of the nearest neighbors to classify a data point. In this method, the predicted class for a new data point is often the mode value obtained from the class labels of its neighbors. Thus, the KNN algorithm classifies a data point based on its neighboring points, and this equation explains this classification process.

$$y = \frac{1}{k} \sum_{i=1}^{k} y_i \tag{13}$$

In Equation 13, it represents the classification process of the K-Nearest Neighbors (KNN) algorithm. While "y" denotes the predicted class of a new data point, "y_i" indicates the class labels of neighboring data points. "k" specifies the number of neighbors used. This equation is a fundamental step in the classification process of the KNN algorithm. The predicted class for a new data point is determined by taking the average of the class labels of its k nearest neighbors. Thus, the data point adopts the predicted class based on the class labels of its neighbors. This equation expresses a simple yet effective classification method of the KNN algorithm.

Gradient Boosting is a powerful machine learning technique for both classification and regression problems. This method constructs a strong predictor by combining many weak predictors. Each weak predictor focuses on correcting the errors of previous predictions. Using an optimization algorithm called gradient descent, Gradient Boosting attempts to minimize these errors. This process continues with the addition of a new weak predictor at each step, gradually reducing the model's errors and making more accurate predictions. One advantage of Gradient Boosting is its ability to combine predictors of different types, usually decision trees, which enhances its capability to learn different data structures and relationships. However, it's crucial to properly adjust the hyperparameters of Gradient Boosting, such as learning rate, tree depth, and number of trees, or else issues like overfitting or longer training times may arise (Bentéjac *et al.* 2021).

$$F_0(x) = 0 \tag{14}$$

In Equation 14, it represents the initial prediction in the Gradient Boosting algorithm. "$F_0(x)$" represents the initial prediction of a new data point, and this initial value is zero. The Gradient Boosting algorithm constructs a prediction model by sequentially adding weak predictors. Initially, the prediction model starts at zero. This equation specifies the beginning of the process for building the prediction model in the Gradient Boosting algorithm.

$$F_m(x) = F_{m-1}(x) + \rho * h_m(x) \tag{15}$$

In Equation 15, "$F_m(x)$" represents the prediction of the new model, while "$F_{m-1}(x)$" denotes the prediction of the previous model, and "$h_m(x)$" represents the m-th weak predictor. "ρ" indicates the learning rate. The Gradient Boosting algorithm uses this equation when adding the next weak predictor to the current prediction model. In other words, in each iteration, the predictions of the current model are updated by adding the predictions of the new predictor multiplied by the learning rate. This way, the algorithm controls the effect of the predictor added in the next step. This equation explains the process of iteratively improving the prediction model in the Gradient Boosting algorithm.

Explainable Artificial Intelligence (XAI) is a branch developed to understand and explain the decisions and predictions of machine learning and artificial intelligence models. XAI emerges from the effort to interpret the inner workings of complex models in a way that is more suitable for human understanding. These techniques contribute to addressing significant issues such as increasing the model's reliability by explaining why and how decisions are made, identifying errors, and addressing fairness and ethical concerns (Ali *et al.* 2023). XAI encompasses various techniques that can help understand the features, variables, and relationships underlying a model's predictions.

These include assessing feature importance, visualizing prediction boundaries, providing instance-based explanations, and analyzing interactions between features. However, XAI methods can themselves be complex, often depending on the complexity of the model, and it's crucial to strike a balance between explainability and model performance.

SHAP (Shapley Additive Explanations), an explainable artificial intelligence (XAI) technique that uses Shapley values to explain the contribution of each feature to model predictions. Shapley values originate from cooperative game theory and estimate the contribution of a feature to a model prediction by considering its value when combined with other features. SHAP is commonly used to make complex machine learning models (such as deep learning or gradient boosting) interpretable. This technique measures the impact of each feature on predictions while also showing how this impact varies for a specific example or observation. Thus, it provides a detailed understanding of why and how a particular prediction was made. SHAP can be used for tasks such as evaluating feature importance, examining interactions between features, and explaining how each feature contributes to model predictions. However, SHAP values can be challenging to interpret, and they can be computationally expensive when working with large datasets or complex models (Das and Rad 2020).

$$\phi_i(f) = \frac{1}{N!} \sum_{\pi} \Big[f(x_{\pi(i)}) - f(x_{\pi}) \Big] \tag{16}$$

Equation 16 represents an explanation method used in the SHAP (SHapley Additive exPlanations) algorithm to measure the contributions of features to the model prediction. "$\phi_i(f)$" represents the contributions of different features, while "f" denotes the model prediction, "N" is the number of data points, "π" is a permutation of data points, "$(x_{\pi(i)})$" represents the i-th data point in a specific permutation, and "x_{π}" denotes the permutation itself. This equation considers all permutations of data points to calculate the contribution of each feature to the model prediction. The SHAP algorithm is used to understand complex model predictions and explain the impact of each feature on the prediction.

LIME is a technique used in the field of explainable artificial intelligence (XAI) to explain how model predictions are made for a specific example or observation. Regardless of the complexity of the model, LIME makes any machine learning model interpretable. This technique creates a surrogate model to understand which features or variables are influential in making a prediction at a particular point. The surrogate model selects features to mimic the predictions of the original model locally. LIME generates data samples by making random changes around the data point to create the local model, and uses the predictions of each sample in the original model. Finally, by examining the behavior of the local model on these generated samples, LIME explains how the original model made a specific prediction. LIME is particularly useful when working with complex models and in situations where understanding why certain predictions are made is difficult. However, interpreting LIME results can be challenging, and careful parameter tuning may be required to obtain accurate results (Das and Rad 2020).

$$e(x) = \arg\min_{g \in \mathcal{G}} (f, g, \pi_x) + \Omega(g) \tag{17}$$

In Equation 17, an explanation method in the LIME (Local Interpretable Model-agnostic Explanations) algorithm is described. "$e(x)$" represents the explanation of a particular example, while the "arg min" operator denotes the one with the smallest value within a given set. "g" represents the model prediction, "f" is the true function, and "π_x" indicates the weights of other examples around the

sample "x". "$\Omega(g)$" is a term limiting the complexity of the model. This equation is formulated as an optimization problem to find the explanation model that best explains the prediction of a particular example. LIME provides local explanations to understand the decisions of complex machine learning models.

RESULTS

In this study, the Biogeography-Based Optimization (BBO) algorithm was used to predict the "share_global_coal_co2" parameter. BBO is an optimization algorithm inspired by natural biogeographic processes, used to determine complex relationships in the dataset and identify the most important features. In this study, the BBO algorithm was used to predict the "share_global_coal_co2" parameter, and the 20 most effective parameters among other parameters in the dataset were identified. BBO assists in predicting the "share_global_coal_co2" parameter by selecting from various parameters in the dataset, thereby helping to predict it more accurately. Therefore, the BBO algorithm was used to predict the "share_global_coal_co2" parameter and understand the impact of specific features. The 20 most important features identified by the BBO algorithm were fed into machine learning models. They were evaluated using metrics such as Mean Squared Error (MSE) and R-squared Score.

Mean Squared Error (MSE) is a metric used to evaluate predictions made by machine learning algorithms. This measurement calculates the average of the squared differences between the actual values and the predicted values. Lower MSE values indicate that the predictions are closer to the actual values, while higher MSE values indicate that the predictions are farther from the actual values. Therefore, MSE is an important measure used to assess a model's predictive ability.

The R-squared (R^2) score is a metric used to evaluate predictions made by machine learning algorithms. This score determines how well a model fits and explains the data. R-squared measures the proportion of the variance in the dependent variable that is predictable from the independent variables. The values typically range from 0 to 1; the closer it is to 1, the better the model explains the data. However, it can also take negative values, indicating that the model performs worse than a horizontal line. In summary, the R-squared score is a measure of how well a model fits the data.

The evaluation results based on the top 20 features identified by the BBO algorithm are presented in Table 1. The Gradient Boosting algorithm, which gave the lowest Mean Squared Error (MSE) value, has been explained using interpretable artificial intelligence models SHAP and LIME. The explanations are presented in Figures 1 and 2.

In Figure 1, the average impact of various greenhouse gas (GHG) emissions and carbon dioxide (CO_2) sources on the model output is assessed using SHAP values. In the figure, total greenhouse gas emissions (total_ghg) and the share of global cumulative cement CO_2 (share_global_cumulative_cement_co2) stand out as the factors with the highest average impact on the model output. Nitrous oxide (nitrous_oxide) and the share of global CO_2 (share_global_co2) also have significant impacts, while factors such as other industry CO_2 (other_industry_co2) and the share of global cumulative coal CO_2 (share_global_cumulative_coal_co2) have less pronounced effects.

Lower impacts are observed among factors such as cumulative land use change CO_2 (cumulative_luc_co2) and temperature change from GHG (temperature_change_from_ghg). Overall, the figure demonstrates that greenhouse gases and various CO_2 emission sources contribute to model outcomes to varying degrees.

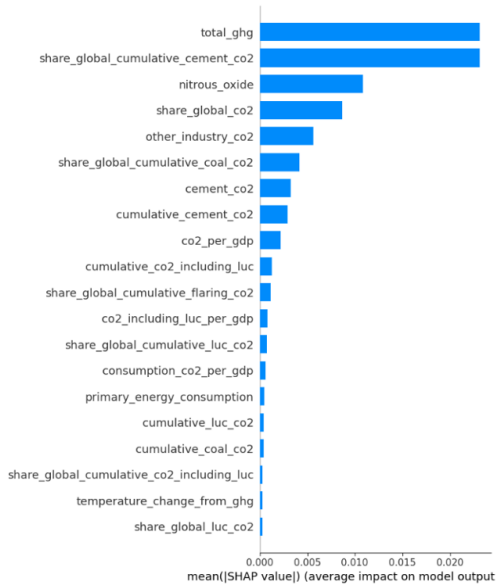

Figure 1 Output of SHAP, an explainable artificial intelligence model

Figure 2 The output of LIME, an explainable artificial intelligence model

In Figure 2, the "GradientBoostingRegressor" algorithm and the interpretable artificial intelligence model "LIME" were used to create predictions for the share of global coal emissions. The predicted value is determined as "-0.20". According to the analysis, the features that the model gives the most importance to are "coal_co2" and "share_global_luc_co", and it is observed that changes in these features have significant effects on the prediction. Additionally, as indicated in the visual, there is a negative correlation with the "share_global" feature's prediction, meaning that as this value increases, the predicted share of global coal emissions decreases

CONCLUSION

Our results demonstrate that the Biogeography-Based Optimization (BBO) algorithm is an effective method for predicting the "share_global_coal_co2" parameter. The BBO algorithm has achieved successful outcomes by utilizing various parameters in the dataset to identify complex relationships and determine significant features. In this study, the 20 most important features identified by the BBO algorithm were integrated into machine learning models to evaluate prediction performance.

Evaluations conducted using metrics such as Mean Squared Error (MSE) and R-squared score have indicated that the Gradient Boosting algorithm provides the lowest MSE value , suggesting that the predictions are closer to the actual values. These results support the use of interpretable artificial intelligence models such as SHAP and LIME to enhance the accuracy of the model.

Table 1 Results of evaluation metrics of machine learning algorithms.

Machine learning	Mean Square Error (MSE)	R-squared Score
ANN	4.017079	-9.101274
XGBoost	0.572459	0.987030
LightGBM	0.372869	0.991552
RF	0.410437	0.990700
SVM	17.084162	0.612933
KNN	26.406132	0.401730
Gradient Boosting	0.347408	0.972128

SHAP and LIME analyses have rendered the model's predictions more understandable. Specifically, the GradientBoostingRegressor algorithm and LIME model utilized to predict the share of global coal emissions have emphasized the effects of specific features such as "coal_co2" and "share_global_luc_co" on the prediction. These findings offer valuable insights that can be utilized in making critical decisions in areas such as energy policies and environmental management strategies.

In conclusion, this study demonstrates that the Biogeography-Based Optimization algorithm is an effective method for predicting the "share_global_coal_co2" parameter. Furthermore, it underscores the importance of utilizing interpretable artificial intelligence models to elucidate and render predictions more comprehensible. These findings serve as a valuable guide for environmental policymakers and energy experts.

Availability of data and material

Not applicable.

Conflicts of interest

The author declares that there is no conflict of interest regarding the publication of this paper.

Ethical standard

The author has no relevant financial or non-financial interests to disclose.

LITERATURE CITED

Ali, S., T. Abuhmed, S. El-Sappagh, K. Muhammad, J. M. Alonso-Moral, *et al.*, 2023 Explainable artificial intelligence (xai): What we know and what is left to attain trustworthy artificial intelligence. Information fusion **99**: 101805.

Bentéjac, C., A. Csörgő, and G. Martínez-Muñoz, 2021 A comparative analysis of gradient boosting algorithms. Artificial Intelligence Review **54**: 1937–1967.

Chen, T. and C. Guestrin, 2016 Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pp. 785–794.

Das, A. and P. Rad, 2020 Opportunities and challenges in explainable artificial intelligence (xai): A survey. arXiv preprint arXiv:2006.11371 .

Delanoë, P., D. Tchuente, and G. Colin, 2023 Method and evaluations of the effective gain of artificial intelligence models for reducing co2 emissions. Journal of Environmental Management **331**: 117261.

Gunn, S. R., 1998 Support vector machines for classification and regression. ISIS technical report **14**: 5–16.

Heo, S., J. Ko, S. Kim, C. Jeong, S. Hwangbo, *et al.*, 2022 Explainable ai-driven net-zero carbon roadmap for petrochemical industry considering stochastic scenarios of remotely sensed offshore wind energy. Journal of Cleaner Production **379**: 134793.

Kramer, O. and O. Kramer, 2013 K-nearest neighbors. In *Dimensionality reduction with unsupervised nearest neighbors*, pp. 13–23, Springer.

Liaw, A. and M. Wiener, 2002 Classification and regression by randomforest. R news **2**: 18–22.

Qerimi, Q. and B. S. Sergi, 2022 The case for global regulation of carbon capture and storage and artificial intelligence for climate change. International Journal of Greenhouse Gas Control **120**: 103757.

Stef, N., H. Başağaoğlu, D. Chakraborty, and S. B. Jabeur, 2023 Does institutional quality affect co2 emissions? evidence from explainable artificial intelligence models. Energy Economics **124**: 106822.

Thanh, H. V., A. Zamanyad, M. Safaei-Farouji, U. Ashraf, and Z. Hemeng, 2022 Application of hybrid artificial intelligent models to predict deliverability of underground natural gas storage sites. Renewable Energy **200**: 169–184.

Yan, H., J. Zhang, N. Zhou, and M. Li, 2020 Application of hybrid artificial intelligence model to predict coal strength alteration during co2 geological sequestration in coal seams. Science of the total environment **711**: 135029.

Zhang, Y., C. Zhu, and Q. Wang, 2020 Lightgbm-based model for metro passenger volume forecasting. IET Intelligent Transport Systems **14**: 1815–1823.

How to cite this article: Ozkurt, C. Environmental Sustainability through AI: A Case Study on CO_2 Emission Prediction. *ADBA Computer Science*, 1(1), 19-25, 2024.