**Carbon Emission Prediction For India**

*Submitted in partial fulfilment of the requirements for the degree of*

**Bachelor of Technology**

in

**Information Technology**

*By*

**DEVANSH KUMAR**

**20BIT0141**

**Under the guidance of**

**Prof. Harshita Patel**

**SCORE**

**VIT, Vellore**



May, 2024

**DECLARATION**

I hereby declare that the thesis entitled “Carbon Emission Prediction For India” submitted by me, for the award of the degree of *Bachelor of Technology in Information Technology* to VIT is a record of bonafide work carried out by me under the supervision of Prof. Harshita Patel

I further declare that the work reported in this thesis has not been submitted, either in part or in full, for the award of any other degree or diploma in this institute or any other institute or university

Place: Vellore

Date: 8th May, 2024

**Signature of the Candidate**

**CERTIFICATE**

This is to certify that the thesis entitled “Carbon Emission Prediction For India” submitted by Devansh Kumar, 20BIT0141, SCORE, VIT, for the award of the degree of *Bachelor of Technology in Information Technology,* is a record of bonafide work carried out by him/her under my supervision during the period, 01.12.2023 to 30.04.2024, as per the VIT code of academic and research ethics

The contents of the report have not been submitted and will not be submitted either part or in full, for the award of any other degree or diploma in this institute or any other institute or university. The thesis fulfils the requirements and regulations of the University and in my opinion meets the necessary standards for submission

Place: Vellore

Date:

**Signature of the Guide**

**Internal Examiner External Examiner**

**Dr. PRABHAVATHY P**

**SCORE**

**ACKNOWLEDGMENTS**

We would like to express our sincere gratitude to Prof. Harshita Patel for her invaluable guidance and support throughout the development of this project. Her expertise and deep understanding of the stock market industry have been instrumental in shaping our research and implementing our machine learning-based system for stock market prediction.

Prof. Patel's commitment to advancing financial technology and her passion for enhancing market analysis have been truly inspiring. Her insights and feedback have greatly influenced the direction of our project, ensuring its relevance and practicality in the field.

We are deeply grateful to Prof. Harshita Patel for her mentorship, patience, and unwavering support. Her expertise and encouragement have been instrumental in the success of this project, and we sincerely appreciate the opportunity to work under her guidance.

**Devansh Kumar**

**Executive Summary**

In 2023 CO2 emission in world was 37.15 billion tons, India contributed around 7.15% around 2.83 billoin tons. Carbon dioxide emissions are the primary driver of global climate change. It is widely recognized that to avoid the worst impacts of climate change, the world needs to urgently reduce emissions.

This thesis explores the prediction of carbon emissions for India using advanced machine learning techniques. The study begins with a comprehensive data collection process, sourcing carbon emission data from reputable sources such as Our World in Data. The dataset includes various factors contributing to carbon emissions, ranging from building emissions to carbon dioxide emissions from electricity and heat.

The study delves into model selection, comparing the performance of three distinct models: Convolutional Neural Network (CNN), Long Short-Term Memory (LSTM) network, and a hybrid CNN-LSTM model. Each model is trained on a portion of the dataset and evaluated using standard metrics like R2 score, Root Mean Square Error (RMSE), and total loss. These findings are crucial for policymakers, environmentalists, and stakeholders to better understand and mitigate carbon emissions in the region. In conclusion, this thesis contributes to the field of environmental science by showcasing the potential of machine learning models in forecasting carbon emissions. The study not only provides valuable predictions but also lays the groundwork for future research in optimizing model performance and addressing environmental challenges.

|  |  |  |  |
| --- | --- | --- | --- |
|  | **CONTENTS** | | **Page no.** |
|  |  | |  |
|  | **Acknowledgement** | | 4 |
|  | **Executive Summary** | | 5 |
|  | **Table of Contents** | | 6 |
|  | **List of Figures** | | 7 |
|  | **List of Tables** | | 7 |
|  | **Abbreviations** | | 8 |
| **1** | **INTRODUCTION** | |  |
|  | 1.1 Objective | | 9 |
|  | 1.2 Motivation | | 10 |
|  | 1.3 Background | | 10 |
| **2** | **LITERATURE REVIEW** | |  |
|  | 2.1 Can Machine Learning be Applied to Carbon Emissions Analysis: An Application to the CO2 Emissions Analysis Using Gaussian Process Regression. (2021) | | 11 |
|  | 2.2 A novel fractional grey Riccati model for carbon emission prediction. | | 11 |
|  | 2.3 Predictions of carbon emission intensity based on factor analysis and an improved extreme learning machine from the perspective of carbon emission efficiency. (2022) | | 12 |
|  | 2.4 Carbon Emission Prediction Model and Analysis in the Yellow River Basin Based on a Machine Learning Method. (2022) | | 12 |
|  | 2.5 Research on the Development of New Energy Vehicles and the Prediction of Carbon Emissions in a Region of Asia. (2023) | | 13 |
|  | 2.6 Grey relational analysis, principal component analysis and forecasting of carbon emissions based on long short-term memory in China. (2019) | | 13 |
|  | 2.7 Carbon Emissions Prediction of Power Industry Based on Extended LMDI and CNN-LSTM. (2023) | | 14 |
| **3** | **OVERVIEW OF PROPOSED SYSTEM** | | 14 |
|  | 3.1 Introduction and Related Concepts | |  |
|  | 3.2 Architecture for the Proposed System | |  |
|  |  | 3.2.1 Carbon Emission Prediction |  |
|  |  | 3.2.2 Data Collection Module |  |
|  |  | 3.2.3 Data Pre-processing Module |  |
|  |  | 3.2.4 Machine Learning Models |  |
|  |  | 3.2.5 Model Training and Evaluation |  |
|  | 3.3 Proposed System Model | |  |
|  | 3.4 Dataset Description | |  |
| **4** | **PROPOSED SYSTEM ANALYSIS AND DESIGN** | |  |
|  | 4.1 Introduction | |  |
|  | 4.2 Requirement Analysis | |  |
|  |  | 4.2.1 S/W requirements |  |
| **5** | **METHODOLOGY** | |  |
|  | 5.1 Data Pre-processing | |  |
|  |  | 5.1.1 Feature Selection |  |
|  |  | 5.1.2 Conversion to NumPy Array |  |
|  |  | 5.1.3 Reshaping Input Data |  |
|  |  | 5.1.4 Train-Split Split |  |
|  | 5.2 Model Building | |  |
|  |  | 5.2.1 Convolutional Neural Network |  |
|  |  | 5.2.2 Long-short Term Memory (LSTM) |  |
|  |  | 5.2.3 CNN-LSTM |  |
|  | 5.3 Model Evaluation | |  |
| **6** | **RESULT AND DISCUSSION** | |  |
|  | 6.1 Graphical representation for carbon emission contributed by different factors | |  |
|  | 6.2 Model comparison | |  |
|  | 6.3 Code extract | |  |
| **7** | **CONCLUSION** | |  |
| **8** | **REFERENCES** | |  |

**List of Figures**

|  |  |  |
| --- | --- | --- |
| **Figure No.** | **Title** | **Page no.** |
| 3.3.1  3.4.1  6.1.1  6.1.2  6.1.3  6.1.4  6.1.5  6.1.6  6.2.1  6.2.2  6.2.3  6.2.4 | Proposed flow of Architecture  Carbon Emission contributed by different sector  Annual Carbon Emission  Per-capita Carbon emission in India  Carbon emission while trading  Carbon emission due to Flaring  Carbon emission by transportation  Carbon emission contribute by different sectors  Actual vs Predicted Graph (CNN)  Actual vs Predicted Graph (LSTM)  Actual vs Predicted Graph (CNN-LSTM)  Predicted value for 2030 | 17  19  25  26  26  26  27  27  28  28  29  29 |

**List of Tables**

|  |  |  |
| --- | --- | --- |
| **Table no.** | **Title** | **Page no.** |
| 6.2.1 | Model Comparison | 27 |

**List of Abbreviations**

|  |  |
| --- | --- |
| **CNN** | Convolutional Neural Networks |
| **LSTM** | Long Short-Term Memory (LSTM) networks |
| **RMSE** | Root Mean Square Error |
| **MSE** | Mean Square Error |
| **R2-score** | R Squared Score |
| **CO2** | Carbon dioxide |
| **FGRM** | Fractional grey Riccati model |
| **MAD** | Mean absolute deviation |
| **STD** | Standard deviation of absolute percentage error |
| **ARIMA** | Auto regressive integrated moving average |
| **PCA** | Principal component analysis |
| **BPNN** | [Back propagation](https://www.sciencedirect.com/topics/earth-and-planetary-sciences/back-propagation) neural network |
| **PCC** | Pearson Correlation Coefficient method |
| **LDMI** | Logarithmic Mean Divisia Index |
| **MAPE** | The mean absolute percentage error |

1. **INTRODUCTION** 
   1. **Objective**

The primary objective of this report is to investigate and develop predictive models for carbon emissions in India using advanced machine learning techniques. Firstly, the report endeavors to curate a comprehensive dataset of carbon emission factors specific to India, drawn from reputable sources such as Our World in Data. This dataset serves as the foundation for subsequent model development and evaluation.

Secondly, the study seeks to explore and compare the performance of various machine learning models, including Convolutional Neural Networks (CNNs), Long Short-Term Memory (LSTM) networks, and hybrid CNN-LSTM models, in predicting carbon emissions. By leveraging state-of-the-art methodologies, the research endeavors to identify the most effective model architecture for accurate and reliable prediction of carbon emissions in the Indian context.

Furthermore, the report aims to rigorously evaluate the performance of each model using standard evaluation metrics such as R2 score, Root Mean Square Error (RMSE), and total loss. This quantitative assessment provides valuable insights into the strengths and limitations of each model, facilitating informed decision-making and policy formulation. Ultimately, the overarching goal of this study is to provide actionable insights and recommendations to policymakers, environmentalists, and stakeholders involved in mitigating carbon emissions in India. By elucidating the complex dynamics of carbon emission prediction and fostering interdisciplinary collaboration, this report aims to catalyze efforts towards a more sustainable and environmentally conscious future for India.

* 1. **MOTIVATION**

The motivation behind this research stems from the urgent need to address the pressing environmental challenges posed by carbon emissions in India. With the country being one of the world's largest contributors to greenhouse gas emissions, there exists a critical imperative to develop effective strategies for mitigating carbon emissions and fostering sustainable development. Furthermore, the adverse impacts of climate change, including rising temperatures, extreme weather events, and ecological disruptions, underscore the necessity for proactive measures to curb carbon emissions. By leveraging advanced machine learning techniques, this research endeavors to contribute to the collective efforts aimed at understanding, predicting, and ultimately reducing carbon emissions in India. The findings of this study hold significant implications for policymakers, environmentalists, and stakeholders, offering valuable insights into the complex dynamics of carbon emissions and guiding evidence-based decision-making towards a greener and more sustainable future.

* 1. **BACKGROUND**

India, as one of the world's fastest-growing economies, faces a significant challenge in balancing economic development with environmental sustainability. With a burgeoning population and rapid industrialization, the country's energy demands have soared, leading to a substantial increase in carbon emissions. According to the Global Carbon Project, India ranks third globally in terms of carbon dioxide emissions, trailing only behind China and the United States. The nation's heavy reliance on fossil fuels, coupled with inefficiencies in energy production and consumption, has exacerbated its carbon footprint, contributing to air pollution, climate change, and environmental degradation. Despite these endeavors, significant challenges persist, necessitating further research and action to achieve sustainable development goals while minimizing environmental impacts.

1. **LITERATURE SURVEY** 
   1. Can Machine Learning be Applied to Carbon Emissions Analysis: An Application to the CO2 Emissions Analysis Using Gaussian Process Regression. (2021)

A nonparametric kernel prediction algorithm in machine learning is applied to predict CO2 emissions. Classical linear regression, w(x) is deterministic whereas the noise term is random. In Gaussian process regression, it is assumed to be random and follows a Gaussian process. A literature review was conducted so that five independent variables; economic growth, energy consumption, population, industrialization, and income, have been identified. Comparison of actual value and prediction of CO2 emissions between the selected models. (A) Classical Least Squares. (B) Robust Least Squares. (C) Exponential GPR

Results showed that the GPR method can give the most accurate predictions on CO2 emissions.

The preciseness and exactitude of the prediction of the exponential GPR were compared and discussed with the classical least squares and the robust least-squares model. Based on the outcome of the whole study, it is proved that the Gaussian progress regression algorithms can give the most accurate predictions on CO2 emissions compared with the other two traditional models.

* 1. A novel fractional grey Riccati model for carbon emission prediction. (2021)

Paper proposes a novel fractional grey Riccati model (FGRM(1,1) model), which combines the Environmental Kuznets Curve hypothesis and differential information principle. The least-squares parameter estimation and mathematics analytical methods are utilized to obtain model parameters and the discrete response function, and the bare bone fireworks algorithm is introduced and designed to obtain the optimal fractional order.

The results show that the FGRM(1,1) model demonstrates better estimation in all cases and efficiency in short-term carbon emission forecasting.

Countries all will reduce their carbon emissions gradually in future and the carbon emissions under current policies are estimated to be 0.31% (Japan), 4.52% (U.S.) and 4.49% (China) below 2020 levels by 2025, respectively. MAPE, mean absolute deviation (MAD), root mean square error (RMSE), and standard deviation of absolute percentage error (STD), are used to validate the effectiveness of the model.

* 1. Predictions of carbon emission intensity based on factor analysis and an improved extreme learning machine from the perspective of carbon emission efficiency. (2022)

Paper starts from the perspective of carbon emission efficiency, applies stochastic frontier analysis to screen the factors influencing carbon intensity, and constructs a model for predicting carbon emission intensity based on factor analysis and an extreme learning machine.

The results suggest that, first, there is a high correlation between carbon emission efficiency and carbon emission intensity. Second, the level of economic development, industrial structure, urbanization level, and government intervention all promote a reduction in carbon emission intensity. The structure of energy consumption and dependence on foreign trade restrain reductions in carbon emission intensity.

SFA model based on the transcendental logarithmic production function, the carbon emission efficiency and its influencing factors were measured and screened for 30 provinces in China.

* 1. Carbon Emission Prediction Model and Analysis in the Yellow River Basin Based on a Machine Learning Method. (2022)

Quadratic assignment procedure regression analysis was used to analyze the factors influencing carbon emissions in the Yellow River Basin from the perspective of regional differences. They propose a machine learning prediction model, namely, the long short-term memory network optimized by the sparrow search algorithm and apply it to carbon emission prediction in the Yellow River Basin. The results show an increasing trend in carbon emissions in the Yellow River Basin, with significant inter-provincial differences.

The carbon emission intensity of the Yellow River Basin decreased from 5.187 t/10,000 RMB in 2000 to 1.672 t/10,000 RMB in 2019, showing a gradually decreasing trend.

* 1. Research on the Development of New Energy Vehicles and the Prediction of Carbon Emissions in a Region of Asia. (2023)

Paper utilizes methods such as short-term time series analysis, entropy weight analysis, and the BP neural network model to make reasonable predictions about future market dynamics.

Separately calculated the influence weights of the three indicator factors, namely fuel prices, electricity prices, and exhaust emissions, on new energy vehicles and traditional fuel vehicles. Predicted vehicle ownership for the next three years using the ARIMA model. By plotting the results and using the BP-neural network model. Regression models for carbon emissions in the energy, transportation, industry, and building sectors, and conducting a multiple linear regression model incorporating these sectors, the dependent variable, independent variables, coefficient estimates, and error terms are determined through fitting with training data. The BP neural network model was used to predict the changes in market share of the two vehicle types over the next 3 years based on current sales data of new energy vehicles and conventional fuel vehicles in the country.

* 1. Grey relational analysis, principal component analysis and forecasting of carbon emissions based on long short-term memory in China. (2019)

This paper presents sixteen potential influencing factors and uses grey relational analysis to identify the factors that have a strong correlation with carbon emissions. The principal component analysis (PCA) is used to extract the four principal components, which reduce the redundancy of the input data. The long short-term memory (LSTM) method is established to predict carbon emissions in China. We use [back propagation](https://www.sciencedirect.com/topics/earth-and-planetary-sciences/back-propagation) neural network (BPNN) and Gaussian process regression (GPR) to compare LSTM method. The simulation results show that the prediction accuracy of carbon emissions based on LSTM is better than that of BPNN and GPR, indicating the effectiveness of PCA and LSTM in prediction of carbon emissions.

* 1. Carbon Emissions Prediction of Power Industry Based on Extended LMDI and CNN-LSTM. (2023)

This paper takes the data of Hunan Province in 1995–2020 as an example. Pearson Correlation Coefficient method (PCCs) and the extended Logarithmic Mean Divisia Index (LMDI) model.a carbon emission prediction model founded on Convolutional Neural Network (CNN)-Long Short-Term Memory (LSTM) is proposed.

Total population, electricity consumption, thermal power generation, per GDP, urbanization rate, proportion of primary industry, proportion of secondary industry and proportion of thermal power generation are strongly related to carbon emissions from power industry, so they are used as the input of the prediction model.

The results indicate that CNN-LSTM reveals an outstanding prediction performance compared with other approaches. In addition, according to the results of scenario analysis, it is found that the adjustment of industrial structure and power generation structure can effectively reduce carbon emissions. R2 and MAPE are 0.983 and 0.0293, With the growth of population, urbanization rate, and per GDP.

1. **OVERVIEW OF PROPOSED SYSTEM** 
   1. **Introduction and Related Concepts**

In today's world, the pressing challenges of climate change and environmental sustainability demand innovative solutions that integrate advanced technologies with comprehensive data analysis. Our proposed system aims to address these challenges by leveraging machine learning techniques to predict carbon emissions, with a specific focus on the context of India. By developing a robust predictive model, we seek to provide policymakers, researchers, and stakeholders with valuable insights into the factors influencing carbon emissions and facilitate evidence-based decision-making for sustainable development.

* 1. **Architecture for the Proposed System** 
     1. Carbon Emissions Prediction:

Carbon emissions prediction involves the use of statistical models and machine learning algorithms to forecast the amount of carbon dioxide (CO₂) and other greenhouse gases emitted by various sources, such as industries, transportation, and energy production. These predictions are crucial for understanding the environmental impact of human activities and formulating strategies to mitigate climate change

* + 1. Data Collection Module**:**

The data collection module retrieves diverse datasets from multiple sources, including government databases, research institutions, and environmental monitoring agencies. These datasets contain information on carbon emissions, economic indicators, population demographics, land use, energy consumption, and other relevant factors.

* + 1. Data Preprocessing Module:

The data preprocessing module performs essential preprocessing tasks to clean, normalize, and transform the raw datasets into a suitable format for model training. This includes handling missing values, encoding categorical variables, scaling features, and extracting relevant features for analysis.

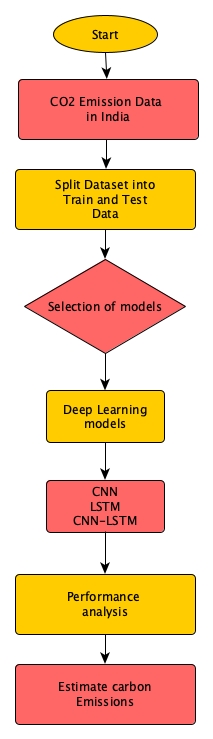
* + 1. Machine Learning Models:

The core of the system consists of machine learning models responsible for predicting carbon emissions based on historical data and contextual factors. These models employ various techniques, including regression, deep learning, and ensemble methods, to capture complex patterns and relationships in the data.

* + 1. Model Training and Evaluation:

The model training and evaluation pipeline train machine learning models using historical emission data and evaluate their performance using standard metrics such as R² score, mean squared error (MSE), and root mean squared error (RMSE). Cross-validation techniques are employed to ensure the robustness and generalization of the models.

* 1. **Proposed System Model**



* + 1. Figure: Proposed flow of Architecture
  1. **Dataset Description**

The dataset utilized for the carbon emission prediction comprises various parameters related to carbon dioxide emissions across different sectors. The parameters include:

* + 1. Carbon dioxide emissions from buildings: This parameter represents the amount of carbon dioxide emissions attributed to residential and commercial buildings.
    2. Carbon dioxide emissions from industry: It denotes the volume of carbon dioxide emissions generated by industrial activities.
    3. Carbon dioxide emissions from land use change and forestry: This parameter accounts for the carbon dioxide emissions and removals associated with changes in land use, such as deforestation and afforestation.
    4. Carbon dioxide emissions from other fuel combustion: It reflects the carbon dioxide emissions resulting from the combustion of fuels other than those used in buildings, industry.
    5. Carbon dioxide emissions from transport: This parameter quantifies the carbon dioxide emissions produced by various modes of transportation, including road, rail, air, and marine transport.
    6. Carbon dioxide emissions from manufacturing and construction: It represents the carbon dioxide emissions generated specifically by manufacturing processes and construction activities.
    7. Carbon dioxide emissions from electricity and heat: This parameter signifies the carbon dioxide emissions produced by the generation of electricity and heat, primarily from fossil fuel-based power plants.
    8. Annual CO₂ emissions (per capita): It denotes the annual carbon dioxide emissions per capita, providing insights into the average emissions generated by individuals within a population.
    9. Annual CO₂ emissions embedded in trade: This parameter accounts for the carbon dioxide emissions embedded in traded goods and services, reflecting the carbon footprint associated with consumption patterns.

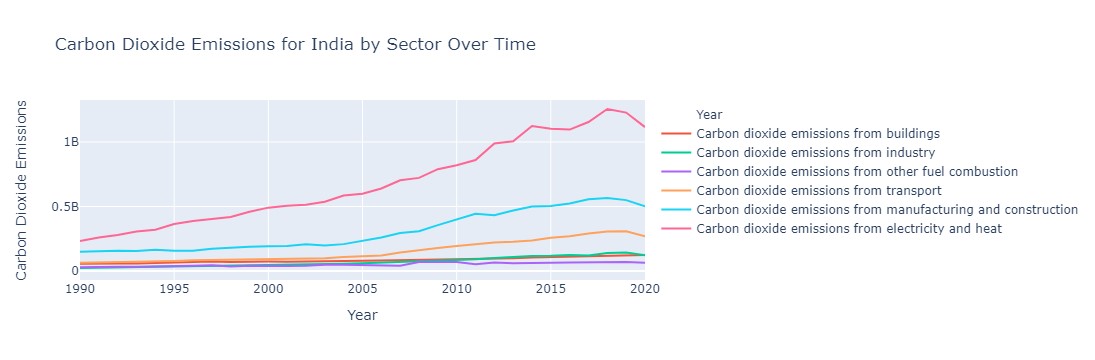


Figure 3.4.1 Carbon Emission contributed by different sector

1. **PROPOSED SYSTEM ANALYSIS AND DESIGN**
   1. **Introduction**

In recent years, the issue of climate change has garnered significant attention worldwide due to its profound impact on the environment, society, and economy. Among the various factors contributing to climate change, carbon dioxide (CO₂) emissions play a central role, primarily stemming from human activities such as industrial processes, transportation, and land use changes. As a result, accurately predicting and monitoring CO₂ emissions has become imperative for understanding climate dynamics, devising effective mitigation strategies, and ensuring sustainable development. To address this pressing need, this research endeavors to develop predictive models for estimating carbon dioxide emissions across different sectors. Leveraging datasets sourced from reputable repositories such as Our World in Data, the study focuses on parameters related to CO₂ emissions from buildings, industry, land use change, transportation, manufacturing, and electricity generation. These parameters provide comprehensive insights into the various sources and drivers of CO₂ emissions, facilitating a nuanced understanding of emission patterns and trends.

The methodology employed in this study encompasses the utilization of advanced machine learning algorithms, including Convolutional Neural Networks (CNN), Long Short-Term Memory (LSTM) networks, and their combination (CNN-LSTM). By harnessing the power of these algorithms, the research aims to harness the temporal and spatial relationships inherent in the data to improve the accuracy of carbon emission predictions.

Furthermore, the evaluation of the predictive models incorporates metrics such as the coefficient of determination (R² score), Mean Squared Error (MSE), and Root Mean Squared Error (RMSE). These metrics provide quantitative assessments of model performance, enabling comparisons between different modeling approaches and guiding the selection of the most effective predictive framework.

Overall, this research endeavors to contribute to the advancement of carbon emission prediction methodologies, thereby facilitating informed decision-making, policy formulation, and climate action initiatives aimed at mitigating the adverse effects of climate change. Through robust modeling techniques and comprehensive evaluation methodologies, the study seeks to enhance our understanding of CO₂ emissions dynamics and pave the way for a more sustainable and resilient future.

* 1. **Requirement Analysis** 
     1. S/W Requirements

The various softwares/libraries used so far are as follows:

● TextPreprocessing

● ML/DL Algorithms

● CNN

● LSTM

● CNN-LSTM

● Artificial neural networks /MLP

● Visualization Dashboard

● Pyplot

1. **Methodology**
   1. Data Preprocessing

The dataset was preprocessed to handle missing values and prepare it for modeling. The features included various carbon emission metrics from different sectors, such as industry, transport, and land use change. The target variable was the annual CO₂ emissions.

* + 1. Feature Selection

The feature matrix X was constructed by dropping the target variable 'Annual CO₂ emissions' from the original dataset. This step ensures that the features used for modeling do not include the target variable.

* + 1. Conversion to Numpy Array

Both the feature matrix X and the target variable y were converted from DataFrame objects to numpy arrays using the to\_numpy() function. This conversion is necessary to work with TensorFlow models, as they require input data in numpy array format.

* + 1. Reshaping Input Data

The feature matrix X was reshaped into a 3D array to match the expected input shape of the Conv1D and LSTM layers in the model. The reshaping was done to add an additional dimension for the number of features per timestep. The reshaping operation is crucial for feeding the data into the model correctly.

* + 1. Train-Test Split

The dataset was split into training and testing sets using the train\_test\_split function from scikit-learn. This step is essential to assess the model's performance on unseen data.

* 1. Model Building

Three different deep learning architectures were explored for the regression task: Convolutional Neural Network (CNN), Long Short-Term Memory (LSTM), and a combination of CNN and LSTM (CNN-LSTM).

* + 1. Convolutional Neural Network (CNN)

The input data for the CNN model consists of sequential features representing different aspects of carbon emissions, such as emissions from buildings, industry, land use change, forestry, and more. Each feature vector represents a specific time step, capturing the evolution of carbon emissions over time. The CNN model begins with convolutional layers designed to extract spatial patterns and relationships within the sequential data. Each convolutional layer applies a set of learnable filters to the input data, convolving over the feature vectors to produce feature maps. These feature maps capture relevant spatial features at different levels of abstraction.

**Activation Function,** Rectified Linear Unit (ReLU) activation functions are used after each convolutional layer to introduce non-linearity into the model and enable the learning of complex patterns.

**Pooling Layers,** Pooling layers, such as max pooling, may be added to the model to down sample the feature maps and reduce the spatial dimensions. This helps in reducing the computational complexity of the model and extracting the most important features.

**Flattening,** After the convolutional and pooling layers, the feature maps are flattened into a one-dimensional vector to be fed into the subsequent fully connected layers. **Fully Connected Layers:** The flattened features are then passed through fully connected (dense) layers, which perform classification or regression tasks based on the learned features. These layers learn to map the extracted features to the desired output, in this case, predicting carbon emissions for future time steps.

**Dropout:** To prevent overfitting and improve the generalization capability of the model, dropout regularization is applied to the fully connected layers. Dropout randomly drops a fraction of the neurons during training, forcing the model to learn more robust features. The final output layer consists of a single neuron with a linear activation function, which produces the predicted carbon emission values for the next time step.

* + 1. Long Short-Term Memory (LSTM)

The LSTM model begins with one or more LSTM layers, each containing a certain number of memory cells (units). These memory cells have the ability to store information over long periods, allowing the model to capture temporal dependencies in the data. The LSTM layers process the sequential input data and propagate information forward through time. Rectified Linear Unit (ReLU) activation functions are used after each LSTM layer to introduce non-linearity into the model and enable the learning of complex temporal patterns.

In the first LSTM layer, the parameter return\_sequences is set to True, indicating that the layer should return the full sequence of output values rather than just the last output. This allows subsequent layers to receive the entire sequence of LSTM outputs, facilitating better feature learning.

**Dense Layers:** After the LSTM layers, the output sequences are passed through one or more dense (fully connected) layers. These layers perform classification or regression tasks based on the learned temporal features. Each dense layer applies a set of learnable weights to the input data and applies an activation function to produce the output. **Dropout** To prevent overfitting and improve the generalization capability of the model, dropout regularization is applied to the dense layers. Dropout randomly drops a fraction of the neurons during training, forcing the model to learn more robust features. **Output Layer** The final output layer consists of a single neuron with a linear activation function, which produces the predicted carbon emission values for the next time step.

* + 1. CNN-LSTM

The CNN-LSTM model combines the strengths of Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks to effectively capture spatial and temporal dependencies in sequential data. In the context of carbon emission prediction, this hybrid architecture offers a powerful approach to analyze both the spatial distribution and temporal trends of emissions.

The input data for the CNN-LSTM model consists of sequential features representing different aspects of carbon emissions, such as emissions from buildings, industry, land use change, forestry, etc. Each feature vector represents a specific time step, capturing the evolution of carbon emissions over time.

The CNN-LSTM model starts with one or more Conv1D layers, which perform 1-dimensional convolution operations on the input data. These layers extract spatial patterns from the sequential input features. The first Conv1D layer applies a set of filters to the input data to detect local patterns, while subsequent layers learn higher-level representations. Rectified Linear Unit (ReLU) activation functions are used after each Conv1D layer to introduce non-linearity into the model and enable the learning of complex spatial features. MaxPooling1D layers are applied after each Conv1D layer to downsample the spatial dimensions of the feature maps. This helps reduce computational complexity and focuses on the most relevant spatial features. After the CNN layers, the output feature maps are passed to one or more LSTM layers. These layers capture temporal dependencies in the data and maintain information over longer sequences. The LSTM layers process the sequential feature maps and propagate information forward through time. ReLU activation functions are again applied after each LSTM layer to introduce non-linearity into the temporal modeling process and facilitate the learning of complex temporal patterns. Following the LSTM layers, one or more dense (fully connected) layers are added to perform classification or regression tasks based on the learned spatial and temporal features. These layers integrate spatial and temporal information and produce the final predictions. The final output layer consists of a single neuron with a linear activation function, which produces the predicted carbon emission values for the next time step.

* 1. Model Evaluation

Each model was trained on the training dataset and evaluated on the test dataset using the following metrics,

R2 Score, Measures the proportion of the variance in the target variable explained by the model.

Root Mean Squared Error (RMSE), Represents the standard deviation of the residuals, indicating the average deviation of predicted values from actual values.

Total Loss, Calculated as the Mean Squared Error (MSE) loss during model training.

1. **RESULTS AND DISCUSSION** 
   1. Graphical representation for carbon emission contributed by different factors

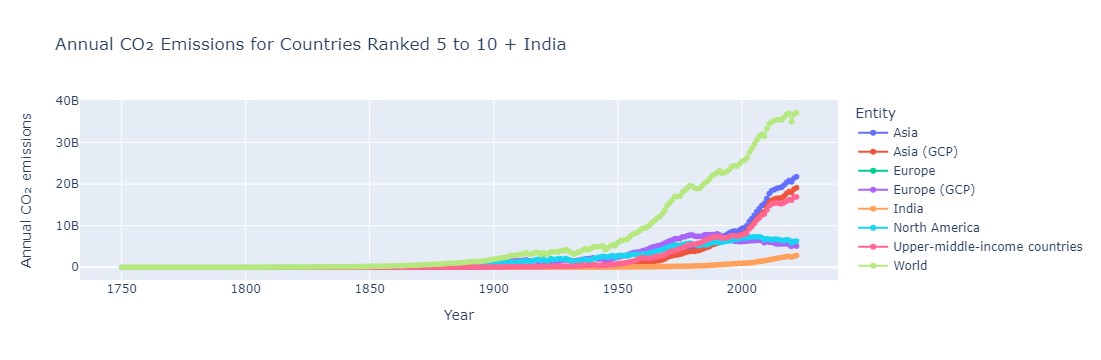


Figure 6.1.1: Annual Carbon Emission

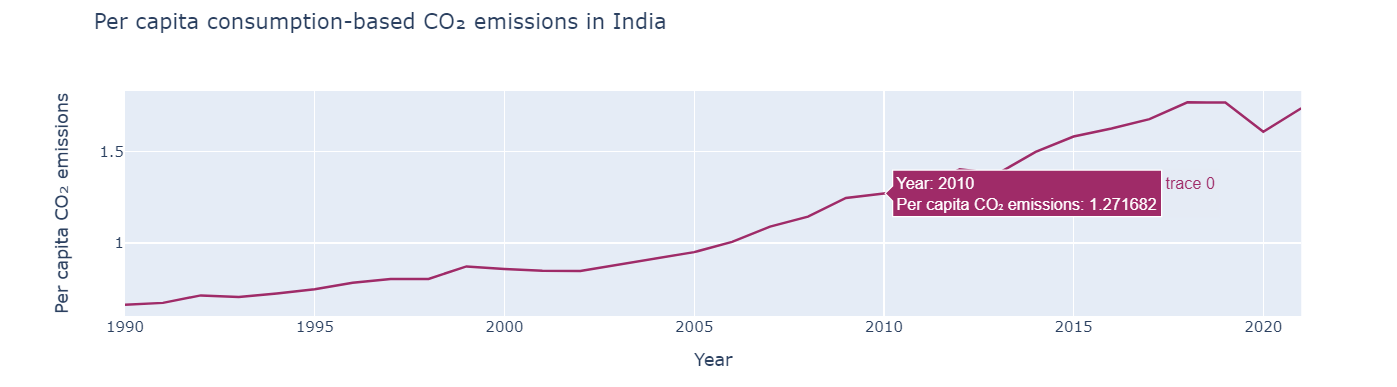


Figure 6.1.2: Per-capita Carbon emission in India

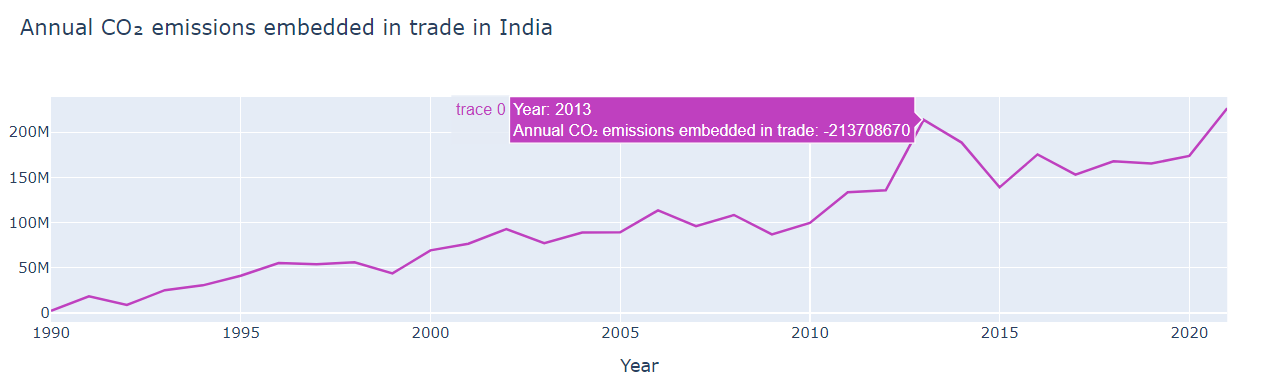


Figure 6.1.3: Carbon emission while trading

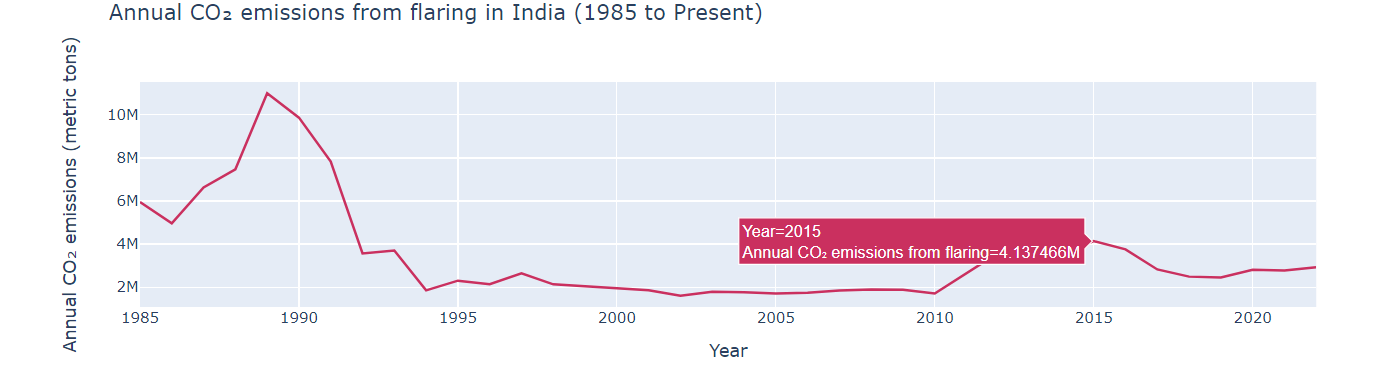


Figure 6.1.4: Carbon emission due to Flaring

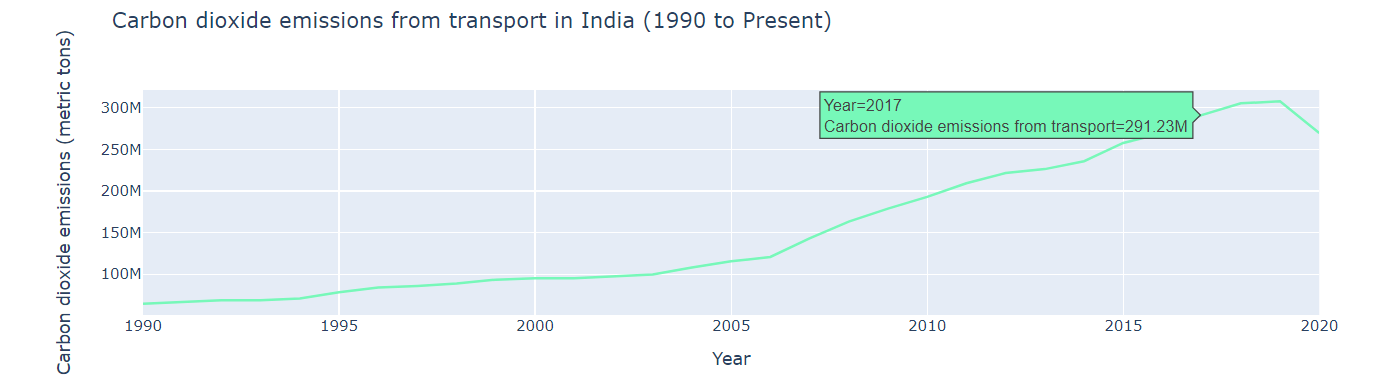


Figure 6.1.5: Carbon emission by transportation

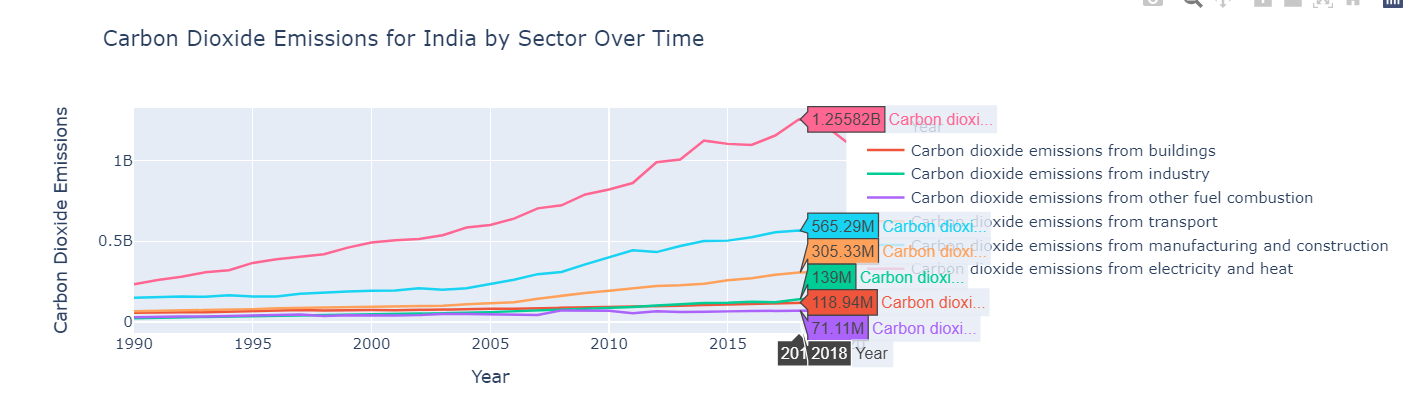


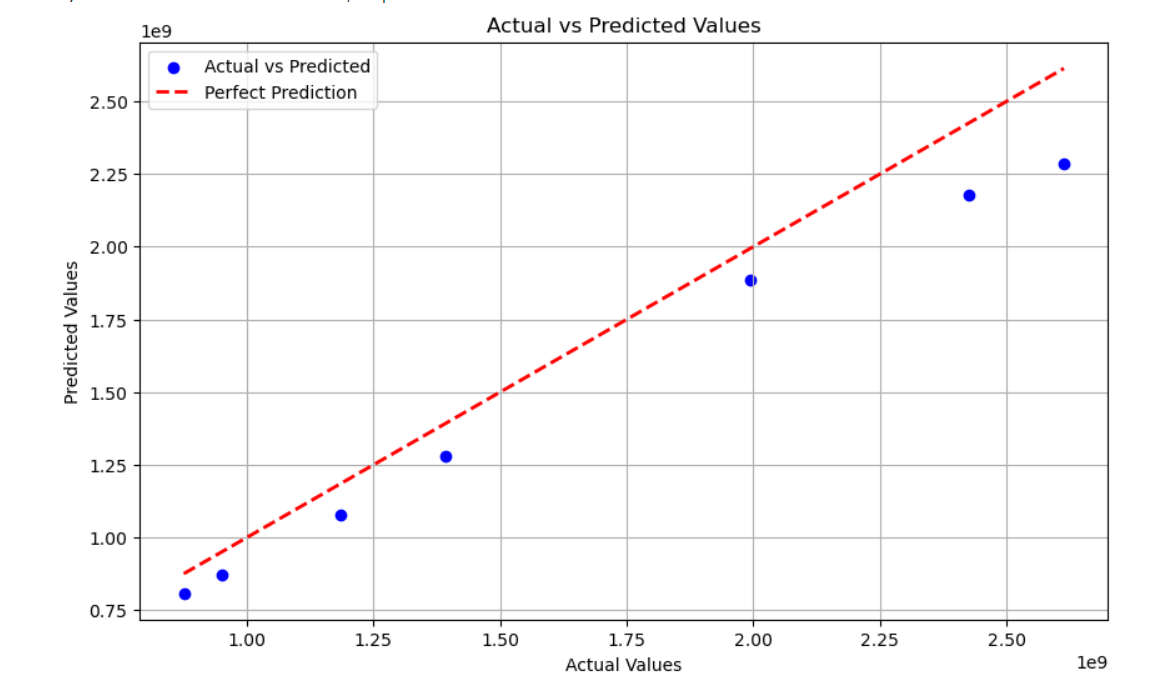
Figure 6.1.6: Carbon emission contribute by different sectors

* 1. Model Comparison

Table 6.2.1: Model Comparison

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Model | R2-SCORE | RMSE | Test Loss | Test MAE |
| CNN | 92.8358 | 175757624.741 | 3.08907555 | 150476720.0 |
| LSTM | 54.0492955 | 445120503.561 | 1.9813225302 | 332129120.0 |
| CNN-LSTM | 95.05133674 | 146074854.921 | 2.133786353 | 119708032.0 |

CNN predicted graph:

Figure 6.2.1: Actual vs Predicted Graph (CNN)

LSTM predicted graph:

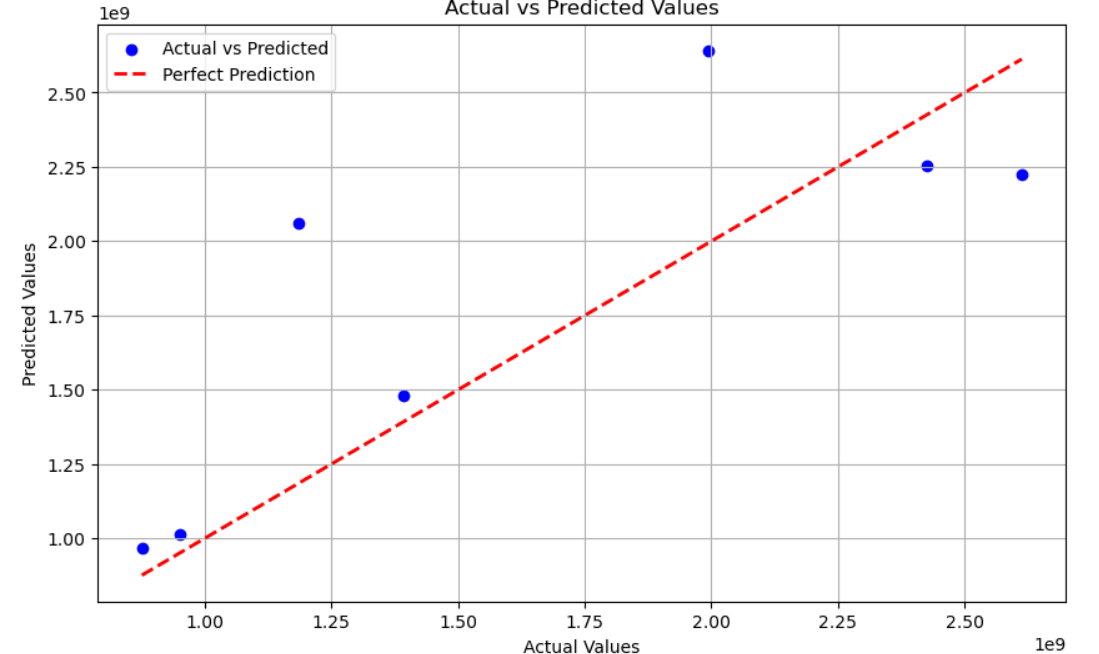


Figure 6.2.2: Actual vs Predicted (LSTM)

CNN-LSTM predicted graph:

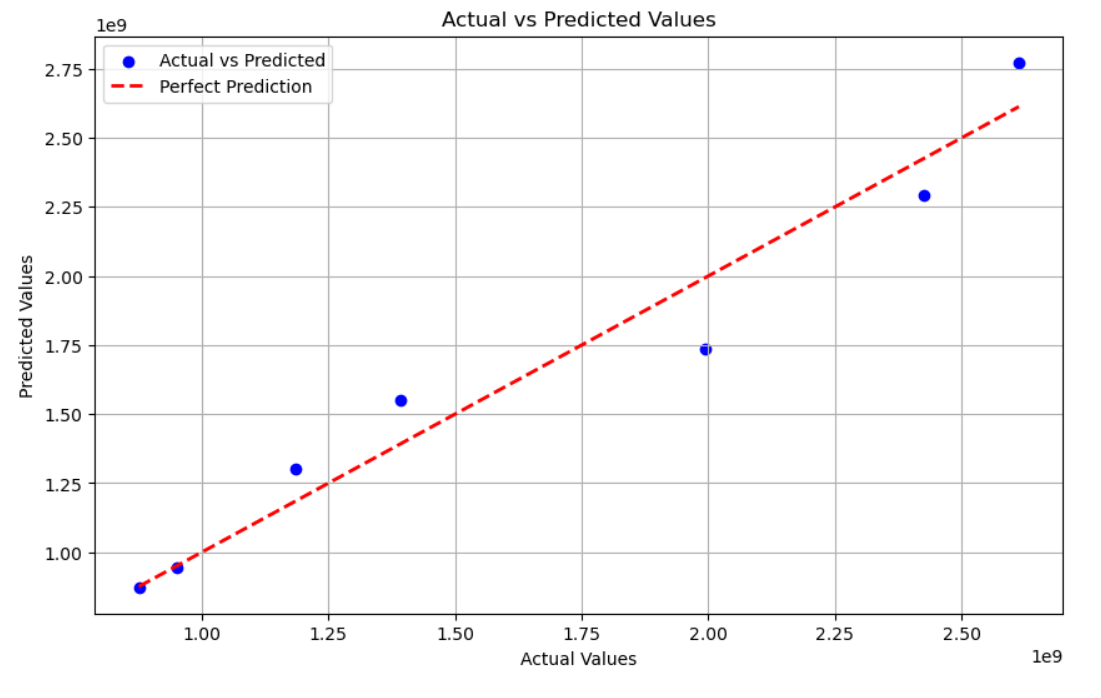


Figure 6.2.3: Actual vs Predicted (CNN-LSTM)

Predicted value by CNN-LSTM for year 2030:



Figure 6.2.4: Predicted value for 2030

* 1. Code Extract

Training Model:

from sklearn.model\_selection import train\_test\_split

# Define features (X) and target variable (y)

# Features remain the same as before

features = ['Carbon dioxide emissions from industry',

'Carbon dioxide emissions from land use change and forestry',

'Carbon dioxide emissions from other fuel combustion',

'Carbon dioxide emissions from transport\_x',

'Carbon dioxide emissions from manufacturing and construction',

'Carbon dioxide emissions from electricity and heat',

'Annual CO2 emissions (per capita)',

'Annual CO₂ emissions embedded in trade',

'Annual CO₂ emissions from flaring']

target = 'Annual CO₂ emissions'

X = data[features]

y = data[target]

# Split the data into training and testing sets (80% train, 20% test)

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=42)

# Display the shapes of the training and testing sets

print("Shape of training set (X\_train):", X\_train.shape)

print("Shape of testing set (X\_test):", X\_test.shape)

print("Shape of training set (y\_train):", y\_train.shape)

print("Shape of testing set (y\_test):", y\_test.shape)

Modeling for CNN:

X\_array = X.values

# Split the data into training and testing sets

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X\_array, y, test\_size=0.2, random\_state=42)

# Reshape the input data to match the expected input shape of Conv1D layers

X\_train = X\_train.reshape(X\_train.shape[0], X\_train.shape[1], 1)

X\_test = X\_test.reshape(X\_test.shape[0], X\_test.shape[1], 1)

# Define the CNN model

model = Sequential([

Conv1D(filters=64, kernel\_size=3, activation='relu', input\_shape=(X\_train.shape[1], X\_train.shape[2])),

Conv1D(filters=128, kernel\_size=3, activation='relu'),

Flatten(),

Dense(128, activation='relu'),

Dropout(0.5), # Adding dropout for regularization

Dense(1) # Output layer with single neuron for regression task

])

# Compile the model

model.compile(optimizer='adam', loss='mse', metrics=['mae'])

# Print model summary

model.summary()

# Train the model

history = model.fit(X\_train, y\_train, epochs=50, batch\_size=32, validation\_split=0.1, verbose=1)

# Evaluate the model on the test set

loss, mae = model.evaluate(X\_test, y\_test)

print("Test Loss:", loss)

print("Test MAE:", mae)

# Predict on test data

y\_pred = model.predict(X\_test)

# Calculate R2 score

r2 = r2\_score(y\_test, y\_pred)

print("R2 Score:", r2)

# Calculate RMSE

rmse = np.sqrt(mean\_squared\_error(y\_test, y\_pred))

print("RMSE:", rmse)

Modeling for LSTM:

model = Sequential([

LSTM(64, activation='relu', input\_shape=(X\_train.shape[1], 1), return\_sequences=True),

LSTM(64, activation='relu'),

Dense(128, activation='relu'),

Dropout(0.5), # Adding dropout for regularization

Dense(1) # Output layer with single neuron for regression task

])

# Compile the model

model.compile(optimizer='adam', loss='mse', metrics=['mae'])

# Print model summary

model.summary()

history = model.fit(X\_train, y\_train, epochs=50, batch\_size=32, validation\_split=0.1, verbose=1)

# Evaluate the model on the test set

loss, mae = model.evaluate(X\_test, y\_test)

print("Test Loss:", loss)

print("Test MAE:", mae)

# Predict on test data

y\_pred = model.predict(X\_test)

# Calculate R2 score

r2 = r2\_score(y\_test, y\_pred)

print("R2 Score:", r2)

# Calculate RMSE

rmse = np.sqrt(mean\_squared\_error(y\_test, y\_pred))

print("RMSE:", rmse)

Modeling for CNN-LSTM:

model = Sequential([

Input(shape=input\_shape),

Conv1D(filters=64, kernel\_size=3, activation='relu'),

MaxPooling1D(pool\_size=2, strides=1),

Conv1D(filters=128, kernel\_size=3, activation='relu'),

MaxPooling1D(pool\_size=2, strides=1),

LSTM(100, activation='relu', return\_sequences=True),

LSTM(50, activation='relu'),

Dense(32, activation='relu'),

Dense(1)

])

# Compile the model

model.compile(optimizer='adam', loss='mse', metrics=['mae'])

# Print model summary

model.summary()

# Train the model

history = model.fit(X\_train, y\_train, epochs=50, batch\_size=32, validation\_data=(X\_test, y\_test), verbose=1)

# Evaluate the model

loss, mae = model.evaluate(X\_test, y\_test)

print("Test Loss:", loss)

print("Test MAE:", mae)

# Predict on test data

y\_pred = model.predict(X\_test)

y\_pred = model.predict(X\_test)

# Calculate R2 score

r2 = r2\_score(y\_test, y\_pred)

print("R2 Score:", r2)

# Calculate RMSE

rmse = np.sqrt(mean\_squared\_error(y\_test, y\_pred))

print("RMSE:", rmse)

1. **Conclusion**

The purpose of this study was to explore different neural network architectures for predicting carbon emissions in India. We implemented three models—CNN, LSTM, and a hybrid CNN-LSTM—and compared their performance based on R²-score, RMSE, test loss, and MAE. The findings indicate that the hybrid CNN-LSTM model was the most effective, achieving the highest R²-score and the lowest error metrics among the three models.

The CNN-LSTM model's ability to combine the spatial feature extraction strengths of CNN with the temporal pattern recognition capabilities of LSTM made it particularly well-suited for this task. This approach yielded an R²-score of 95.05133674 and a predicted carbon emissions value for 2030 of 131 billion tons, providing valuable insights for stakeholders involved in policy-making and environmental planning.

The LSTM model, while capable of capturing temporal dependencies, had a lower R²-score and higher error metrics, indicating that it struggled to accurately predict carbon emissions. The CNN model performed better than the LSTM model but was outperformed by the hybrid CNN-LSTM approach, suggesting that integrating both spatial and temporal analysis provides a more comprehensive understanding of carbon emissions data.

The conclusion drawn from this study is that the CNN-LSTM hybrid model is a robust tool for predicting carbon emissions in India. This model can serve as a foundation for future work aimed at addressing climate change and sustainability. It also demonstrates the importance of adopting advanced machine learning techniques in environmental research. Future studies could focus on further refining the CNN-LSTM model and exploring additional features or datasets to improve accuracy and reliability. Additionally, research into using these predictions for proactive policy-making and carbon emission reduction strategies would be a logical next step.

1. **Reference**
   1. Ma, N., Shum, W. Y., Han, T., & Lai, F. (2021). Can machine learning be applied to carbon emissions analysis: an application to the CO2 emissions analysis using Gaussian process regression. *Frontiers in Energy Research*, *9*, 756311.
   2. Gao, Mingyun, et al. "A novel fractional grey Riccati model for carbon emission prediction." *Journal of Cleaner Production* 282 (2021): 124471.
   3. Sun, W., & Huang, C. (2022). Predictions of carbon emission intensity based on factor analysis and an improved extreme learning machine from the perspective of carbon emission efficiency. *Journal of Cleaner Production*, *338*, 130414.
   4. Zhao, J., Kou, L., Wang, H., He, X., Xiong, Z., Liu, C., & Cui, H. (2022). Carbon emission prediction model and analysis in the Yellow River basin based on a machine learning method. *Sustainability*, *14*(10), 6153.
   5. Luo, W., Deng, J., Zhou, T., & Xu, R. (2023, September). Research on the Development of New Energy Vehicles and the Prediction of Carbon Emissions in a Region of Asia. In *2023 International Conference on Network, Multimedia and Information Technology (NMITCON)* (pp. 1-7). IEEE.
   6. Huang, Y., Shen, L., & Liu, H. (2019). Grey relational analysis, principal component analysis and forecasting of carbon emissions based on long short-term memory in China. *Journal of Cleaner Production*, *209*, 415-423.
   7. Shen, M., Xu, J., Shi, Z., Wu, M., & Ye, Y. (2023, September). Carbon Emissions Prediction of Power Industry Based on Extended LMDI and CNN-LSTM. In *2023 International Conference on Power System Technology (PowerCon)* (pp. 1-7). IEEE.