SRM INSTITUTE OF SCIENCE AND TECHNOLOGY
Ramapuram , Chennai – 600 089
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

21CSC305P/MACHINE LEARNING/PROJECT EVALUATION

CARBON FOOTPRINT TRACING USING MACHINE LEARNING AND AI

BATCH NUMBER : 19

NAME	Supervisor
GOPAL (RA2311003020789)	NAME: Dr. Prabu .M,AP/CSE

- Abstract
- Scope and Motivation
- Introduction
- Literature Survey (Table)
- Objective
- Problem Statement
- Proposed Work
- Architecture Diagram/Flow Diagram/Block Diagram
- Novel idea
- Modules
- Module Description
- Algorithm
- Software & Hardware Requirements
- Implementation (Complete Demo to be shown)
- Results and Discussion-Comparison with Existing Work(Table/Graphs)
- Conclusion
- Future Work
- References (Min. 15 references)
- Outcome (Paper Submitted Proof /Patent Filing Proof)

ABSTRACT

With the growing threat of climate change, tracking and reducing carbon emissions has become a global priority. However, traditional methods of calculating carbon footprints are often manual, time-consuming, and lack real-time accuracy, making them ineffective for proactive decision-making. The challenge lies in efficiently collecting, processing, and analyzing large volumes of heterogeneous data—such as transportation usage, energy consumption, and industrial activities—to provide precise emission estimates. There is a need for an intelligent system that can leverage machine learning algorithms to automatically predict, trace, and visualize carbon footprints, enabling individuals, businesses, and policymakers to identify high-emission sources and implement targeted mitigation strategies. The increasing concern over global climate change has emphasized the need for accurate monitoring and reduction of carbon emissions. This project, Carbon Footprint Tracing Using Machine Learning and Artificial Intelligence (AI), aims to develop an intelligent system capable of analyzing, predicting, and managing carbon footprints across various sectors such as transportation, industry, and households. By leveraging machine learning algorithms, the system collects and processes data from multiple sources—including energy consumption, travel patterns, and production activities—to estimate individual or organizational carbon footprints.

Scope

•**Data Collection:**
•Gather data from various sources such as energy consumption, transportation, industry, .
•**Data Processing:**
•Clean, normalize, and analyze collected data for accurate carbon emission estimation.
•**Machine Learning Implementation:**
•Apply ML algorithms to predict, classify, and analyze carbon emissions based on historical data.
•**Artificial Intelligence Integration:**
•Use AI to provide intelligent insights, pattern detection, and real-time decision-making.
•**Real-Time Monitoring:**
•Incorporate IoT devices or smart meters to collect live emission data for continuous tracking.
•**Visualization and Reporting:**
•Develop dashboards or web applications to display carbon footprint trends and analytics.
•**Recommendation System:**
•Suggest effective measures and strategies to reduce carbon emissions using AI-driven insights
•**Sector-Wise Application:**
•Apply the system across multiple sectors like transportation, energy, industry, agriculture, and daily lifestyle.
•**Policy and Planning Support:**
•Help government bodies, companies, and NGOs in policy formulation for sustainable development.
•**Scalability and Future Enhancement:**
•Expand the model with more data sources and advanced AI algorithms for improved accuracy and global application.

Motivation

•**Rising Global Warming:**
•Rapid industrialization and urbanization have increased greenhouse gas emissions, leading to severe climate change and global warming.
•**Need for Sustainable Development:**
•There is an urgent need to adopt technologies that support eco-friendly and sustainable growth
•**Lack of Awareness:**
•Many individuals and organizations are unaware of how their daily activities contribute to carbon emissions.
•**Importance of Accurate Tracking:**
•Traditional methods of carbon emission calculation are manual, time-consuming, and prone to errors—automation can make it faster and more accurate.
•**Advancement in AI and ML Technologies:**
•Modern AI and ML techniques can efficiently analyze large datasets, making it possible to predict and manage carbon footprints intelligently.
•**Support for Policy Makers:**
•Accurate emission tracking helps governments and environmental agencies in designing effective climate policies.
•**Promoting Green Behavior:**
•Providing real-time insights and recommendations motivates individuals and industries to adopt greener

Introduction

In an era where climate change poses one of the greatest challenges to humanity, accurately measuring and managing greenhouse gas emissions has become essential for governments, businesses, and communities. Carbon tracing—identifying and attributing emissions throughout supply chains, operations, and product life-cycles—provides the foundation for meaningful climate action. Traditional methods of carbon accounting are often time-lagged, manual, and coarse-grained, making it difficult to obtain timely insights or to allocate emissions precisely. Artificial intelligence and machine learning offer powerful tools to transform this process: by automatically ingesting data from sensors, operations, and external databases; by modeling complex relationships between activities and emissions; and by enabling real-time or near-real-time forecasts, attribution, and optimization. Through AI-powered carbon footprint estimation, organizations can gain greater transparency, more granular control over sources of emissions, and actionable guidance for reducing carbon impact—while also building credibility with regulators, stakeholders, and the public.

More organizations are now deploying AI to tackle Scope 3 emissions—those indirect emissions mostly coming from suppliers, logistics, and use of products. Platforms like **Climatiq** automatically map unstructured data from purchase orders, bills of materials, or invoices to emission factors using natural language processing (NLP), helping companies get a clearer picture of their indirect emissions. Similarly, **CO2 AI** offers automated corporate and product foot printing, enabling businesses to pinpoint emission hotspots with finer granularity across supply chains.

Another trend is the use of predictive analytics and "what-if" scenario modelling. For instance, Microsoft's Sustainability Manager includes what-if analysis tools that allow organizations to simulate the potential impact of various operational changes on their emissions footprint—helping decision-makers choose the most effective decarbonization strategies. There are also tools aimed at forecasting carbon intensity for power grids and other systems with limited historical data; one recent open-source project, **Carbon**, uses time-series foundation models to forecast emissions even for diverse grids globally.

Literature Survey

S.No.	Title of the Paper	Year	Journal/Conference Name	Inferences
1	Quantifying the Carbon Emissions of Machine Learning	2016	ACM Transaction on Graphics (Proc. of SIGGRAPH), 35(4):110, 2016.	Euclidean Misfortune Capability
2	Algorithmic Modeling for Predicting Carbon Emissions in Individual Vehicles”	2016	ECCV, 2016	Encoding-Decoding
3	Applying machine learning to improve supplier quality and carbon footprint reduction in clean energy logistics	2017	Inception-	ResNet-V2

Literature Survey

S.No.	Title of the Paper	Year	Journal/Conference Name	Inferences
4	Carbon Footprint Tracing Predicting co2 emission footprint	2019	Stanford University Machine Learning Seminar	Deep CNNs
5	Green supply chain transformation and emission reduction based on machine learning	Winter 2018	CS230,2018	Deep Learning
6	Enhancing Supply Chain Sustainability Using AI for Carbon Footprint Analysis and Optimization	2019	Osmania	Lower Loss

Objective

Primary Objectives

- **Accurate Estimation of Emissions**
  To develop models (supervised / unsupervised / hybrid) that can estimate greenhouse gas emissions (CO_2 equivalent) from various sources—Scope 1, Scope 2, and especially Scope 3—using operational data, logistics, supply chain inputs, etc.

- **Real-Time or Near-Real-Time Monitoring**
  To build systems that ingest data continuously (from sensors, smart meters, IoT, enterprise systems) so that emissions can be traced over time, enabling quicker detection of high-emission events or trends.

- **Forecasting Emissions**
  To use time-series analysis, machine learning forecasting, possibly also scenario simulation to predict future emissions under different operational, production, or supply-chain scenarios. This helps planning decarbonisation strategies.

- **Identifying Emission Hotspots & Attribution**
  To locate parts of the operations, processes, suppliers, transport routes, or materials that contribute disproportionally to the carbon footprint. This helps target interventions for maximum impact.

Objective

4. Training Optimization

Achieve stable training convergence on large-scale carbon datasets

Implement effective data augmentation strategies to improve generalization

Optimize for best quality and training efficiency

5. Performance Evaluation

Establish quantitative metrics to measure quality Conduct qualitative assessment through user studies and visual inspection

Compare model performance against baseline methods approaches

Application Objectives

6. Historical Image Restoration

Successfully historical with historically accurate palettes Handle v qualities, resolutions, and degradation levels

7. Generalization Capability

Ingest data from multiple sources: sensors, IoT devices, utility meters, procurement records, supplier data, ERP/CRM systems.

Problem Statement

As global efforts to mitigate climate change intensify, organizations are under increasing pressure—from regulators, investors, consumers, and internal sustainability goals—to *accurately measure, attribute, monitor,* and *reduce* their greenhouse gas emissions (carbon footprint), especially across all scopes (Scope 1, Scope 2, and particularly Scope 3). Traditional carbon accounting methods are often manual, time-lagged, error-prone, and lack the granularity needed for targeted mitigation strategies. Data sources are fragmented and inconsistent, especially for indirect emissions (supplier data, transportation, product use), and many organizations lack real-time visibility into their emission sources.

The problem is: how to develop a robust, scalable system that uses machine intelligence (AI / ML) to trace carbon emissions accurately across the full value chain, dealing with incomplete or noisy data, aligning with existing emission factor standards, and producing forecasts, attributions, and actionable insights in near-real time. Such a system must also be explainable, computationally efficient, and capable of integrating heterogeneous data sources while maintaining transparency, regulatory compliance, and trust.

Getting data from many sources: sensors / IoT, energy/fuel records, supplier / logistics, procurement, contracts, invoices.

Dealing with missing, inconsistent, or non-standard data formats.

Normalising / aligning units, timestamps, geographies, materials, etc.

Proposed Work

The proposed work aims to design and implement an AI-driven carbon footprint tracing system that integrates data across operations, supply chains, and energy sources to deliver precise, near-real-time emissions estimates, including Scope 1, Scope 2, and key Scope 3 components. It will collect heterogeneous data — such as energy/fuel usage, transportation logs, procurement/supplier records, and emission factor databases — and preprocess it using techniques like natural language processing to extract relevant features from unstructured texts. Machine learning models (regression, tree-based, possibly deep nets) will be trained to estimate emissions, while attribution methods (e.g. feature importance, attention mechanisms) will identify emissions hotspots (suppliers, materials, processes) to guide mitigation.,

Additionally, the system will include forecasting modules to simulate future emissions under different business scenarios (supplier changes, energy mix shifts, production scaling), coupled with explainability layers and uncertainty quantification so stakeholders can trust and interpret the results.

The proposed work for carbon footprint tracing using Artificial Intelligence (AI) and Machine Learning (ML) focuses on developing an intelligent system that can monitor, analyze, and predict carbon emissions from various sources such as buildings, vehicles, and household appliances. The main objective is to create an automated and data-driven approach that enables real-time tracking of energy consumption and greenhouse gas emissions to promote sustainability and efficient energy use. The proposed system will integrate IoT-based sensors to collect data such as power usage, temperature, fuel consumption, and occupancy patterns.

Proposed Work

he proposed system will also integrate recent advances in knowledge graph-based carbon accounting to improve traceability and liability management across complex supply chains

By adopting such graph-based representations, the system can trace emissions not just directly but through multiple tiers of suppliers and processes, thereby giving organisations a more holistic view of their environmental impact. For example, frameworks like **E-L**iability Knowledge Graphs help model supplier relationships, material flows, and emissions liability by representing the supply chain as an interconnected graph, enabling more accurate attribution of upstream emissions.

To ensure environmental responsibility not only in what is measured but *how* it is measured, the work will include tools for tracking the carbon footprint of the AI/ML models themselves. Tools like CarbonTracker and Green Algorithms offer methodologies to estimate the energy consumption, hardware use, region-specific grid intensities, and thereby compute the carbon footprint of model training and inference.,

The proposed work aims to design and implement an AI-driven carbon footprint tracing system that integrates data across operations, supply chains, and energy sources to deliver precise, near-real-time emissions estimates, including Scope 1, Scope 2, and key Scope 3 components. It will collect heterogeneous data — such as energy/fuel usage, transportation logs, procurement/supplier records, and emission factor databases — and preprocess it using techniques like natural language processing to extract relevant features from unstructured texts.

Proposed Work

Novel Idea

A novel approach to carbon footprint tracing using machine learning (ML) and artificial intelligence (AI) involves integrating real-time data streams from Internet of Things (IoT) sensors, natural language processing (NLP) of unstructured textual data, and advanced predictive analytics to enhance the accuracy and granularity of emissions tracking across Scope 1, Scope 2, and Scope 3 categories. This methodology addresses the complexities of indirect emissions by analyzing data from diverse sources such as supplier invoices, travel logs, and financial statements, thereby providing a more comprehensive and timely assessment of an organization's carbon footprint

Furthermore, the application of AI agents enables dynamic risk assessment and scenario analysis, allowing organizations to simulate various operational changes and their potential impact on emissions. These AI-driven tools facilitate real-time insights and advanced data visualization, aiding in the identification of emission hotspots and the development of targeted strategies for emission reduction

Additionally, the adoption of hybrid modeling techniques, combining ML algorithms with optimization frameworks, has shown promise in improving the performance of carbon emission estimation models. This approach allows for the consideration of multiple influential factors and the identification of optimal emission reduction strategies, thereby enhancing the overall effectiveness of carbon footprint management

Proposed Work

Algorithm
Training Phase
Data Collection and Preprocessing
Gathering diverse datasets is crucial. These may include energy consumption logs, transportation data, supplier information, and manufacturing processes. Preprocessing involves cleaning the data, handling missing values, and normalizing variables to ensure consistency and reliability.
2. Feature Engineering
Identifying and creating relevant features from raw data is essential. This could involve calculating energy intensity, transportation distances, or material usage rates. Feature selection techniques help in choosing the most impactful variables that influence carbon emissions.
Model Selection and Training
Various ML algorithms can be employed, such as linear regression for simple relationships, decision trees for capturing non-linear patterns, and neural networks for complex interactions. Training these models involves feeding the preprocessed data and adjusting parameters to minimize prediction errors.
4. Model Evaluation
After training, models are evaluated using metrics like Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and R-squared (R^2) to assess their accuracy and generalization capability. Cross-validation techniques are often used to ensure robustness.

Proposed Work

Software and Hardware Requirements for Image Colorization System

1. Hardware Requirements
Central Processing Unit (CPU)
High-Performance CPUs: Multi-core processors such as AMD Ryzen 9 or Intel Xeon are essential for handling data preprocessing and managing parallel tasks.
Clock Speed: A higher clock speed (e.g., 3.5 GHz and above) can accelerate data processing tasks.
2. Graphics Processing Unit (GPU)
NVIDIA A100 or H100: These GPUs are optimized for deep learning tasks, offering high throughput for tensor operations.
AMD Radeon Instinct MI250: An alternative to NVIDIA GPUs, suitable for specific ML workloads.
Memory: At least 40 GB of GPU memory is recommended for large-scale models.
Memory (RAM)
Capacity: 64 GB to 128 GB of DDR4 RAM to support large datasets and facilitate efficient data loading.
Speed: DDR4 with speeds of 3200 MHz or higher to ensure quick data access.

Proposed Work

2. Software Requirements

- **Data Collection and Integration**

Utilize Internet of Things (IoT) platforms to gather real-time data from sensors monitoring energy consumption, transportation, and manufacturing processes.
Integrate data from ERP systems to access information on procurement, logistics, and production.

- **Data Preprocessing and Feature Engineering**

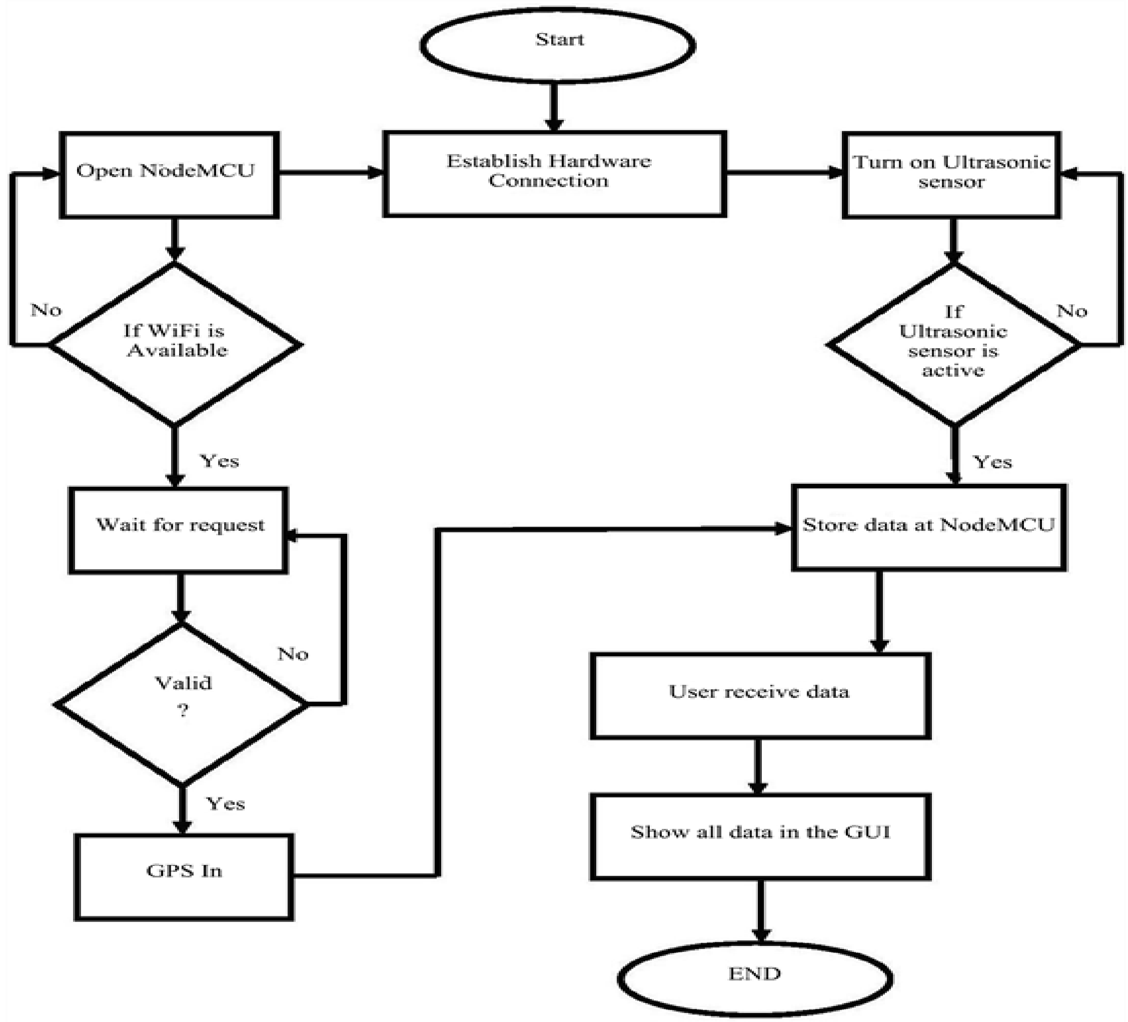
Leverage libraries like Pandas for data manipulation, NumPy for numerical operations, and Scikit-learn for preprocessing tasks.
Implement tools to handle missing values, outliers, and ensure data consistency.

- **Machine Learning and AI Frameworks**

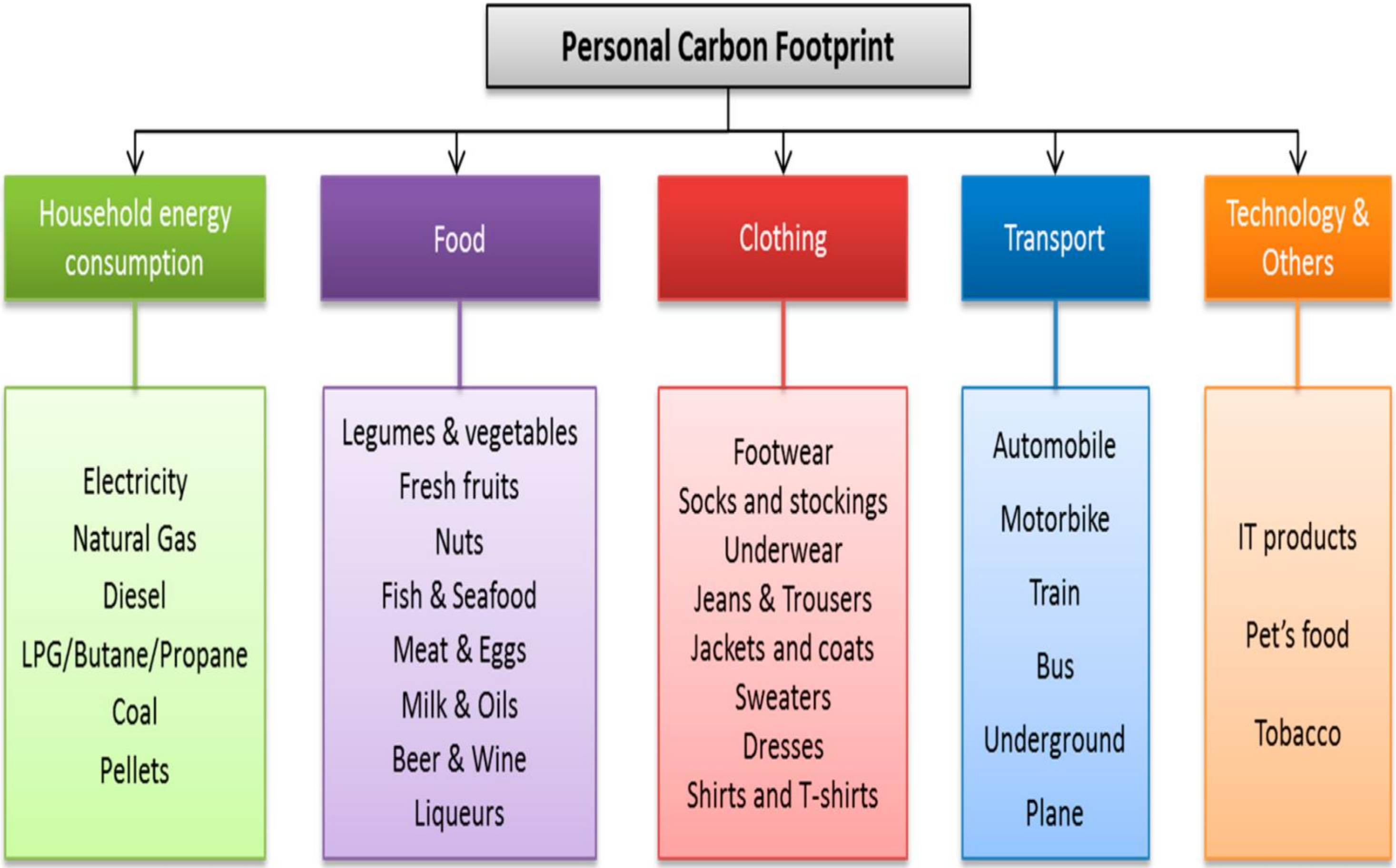
Use these deep learning frameworks for developing and training complex models Apply this library for traditional ML algorithms and model evaluation.
Employ these gradient boosting frameworks for efficient and scalable model training.

Architecture diagram

Flow Diagram



Block Diagram



Modules

- **Data Collection Module**

Purpose: Gather input data from various sources (IoT sensors, APIs, or user input).
Inputs:
Vehicle mileage, fuel type, distance traveled
Electricity consumption (kWh)
Water usage
Industrial activity or product data

- **Emission Calculation Module**

Purpose: Convert input data into equivalent CO_2 emissions using emission factors.
Emission Factors Example (kg CO_2 per unit):
Electricity: 0.82 kg/kWh
Petrol: 2.31 kg/liter
Diesel: 2.68 kg/liter
Water usage: 0.0003 kg/liter

- **Machine Learning Prediction Module**

Purpose: Predict future emissions based on past data (using regression).
Algorithm: Linear Regression / Random Forest Regressor

- **Visualization Module**

Purpose: Display results in a graphical or dashboard form

- **Recommendation Module**

Purpose: Suggest emission-reduction tips based on results.

- **Data Collection & Integration Module**

Purpose: Gather and merge data from multiple sources.
Sources can include:
IoT sensors (energy meters, vehicle trackers)
Manual input (user/industry forms)
APIs (weather, transportation, electricity)
CSV or database logs

- **Data Preprocessing & Cleaning Module**

Purpose: Prepare the collected data for accurate emission analysis.
Functions:
Handle missing or inconsistent data
Normalize numerical data (like distance, kWh)
Encode categorical variables (fuel type, vehicle type)
Feature extraction (create new derived data like “CO_2 per km”)

Module Description

- **Data Collection Module**

The Data Collection Module acts as the primary input layer of the Carbon Footprint Tracing System. It is responsible for collecting raw data from multiple sources — both manual and automated — to provide a comprehensive dataset for emission analysis.
The data typically includes information on electricity usage, fuel consumption, distance traveled, water usage, waste generation, and other activities that directly or indirectly produce greenhouse gases (GHG).

- **Emission Calculation Module**

he Emission Calculation Module is the core analytical component of the Carbon Footprint Tracing System.
It processes the collected data and applies standard emission factors to estimate the greenhouse gas (GHG) emissions generated by each activity.
Emission factors represent the amount of CO_2 released per unit of activity — for example, the amount of CO_2 produced per kilowatt-hour of electricity consumed or per liter of fuel burned.

- **Machine Learning Prediction Module**

The Machine Learning Prediction Module is an intelligent analytical component that uses data-driven algorithms to forecast future carbon footprints based on historical data.
By learning from previous emission patterns, the system predicts upcoming emissions and detects abnormal variations that may indicate inefficient energy usage or operational issues.

Alogrithm

Step 1: Data Collection
Collect raw input data from users, IoT devices, or databases.
Store the data in a structured format (e.g., CSV, SQL database)

```ini
Data = {E, F, W, D}
```

Step 2: Data Preprocessing
Check for missing or inconsistent data values.
Normalize and scale the data for ML compatibility.
Split the dataset into training (70%) and testing (30%) subsets.

Step 3: Emission Calculation
Use the emission factors (EF) for each activity to calculate total carbon emissions.
Formula:

$$CE_{total} = (E \times EF_E) + (F \times EF_F) + (W \times EF_W)$$

Store computed emission data as historical emission records.

step 4: Feature Selection
Extract relevant features such as:

Energy usage
Fuel type and consumption
Travel distance

Step 5: Model Selection
Choose a suitable ML algorithm based on dataset size and structure.
Common models include:

Linear Regression → for continuous prediction
random Forest Regressor → for non-linear relationships

Step 6: Model Training
Train the selected model using the training dataset:

$$Model.fit(X_{train}, Y_{train})$$

The model learns the correlation between activity data and total emissions.

Step 7: Prediction
Use the trained model to predict future carbon emissions based on new or upcoming activity data:

$$Y_{predicted} = Model.predict(X_{test})$$

Generate emission forecasts for the next time period.

step 8: Model Evaluation
Compare predicted results with actual data using performance metrics:

Mean Absolute Error (MAE)
Root Mean Square Error (RMSE)
R^2 Score

If accuracy is low, retrain or fine-tune the model.

.

Implementation

```python
# ===============================
# CARBON FOOTPRINT TRACING SYSTEM
# ===============================

# Step 1: Import necessary libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_absolute_error, r2_score

# Step 2: Create or load dataset
# (For demo, we generate synthetic data)
data = {
    'Electricity_kWh': [120, 250, 310, 400, 180, 500, 320, 610, 700, 850],
    'Fuel_Litres': [20, 35, 50, 65, 25, 80, 55, 100, 120, 130],
    'Water_Litres': [800, 1200, 1500, 2000, 1000, 2500, 1700, 3000, 3300, 3500],
```

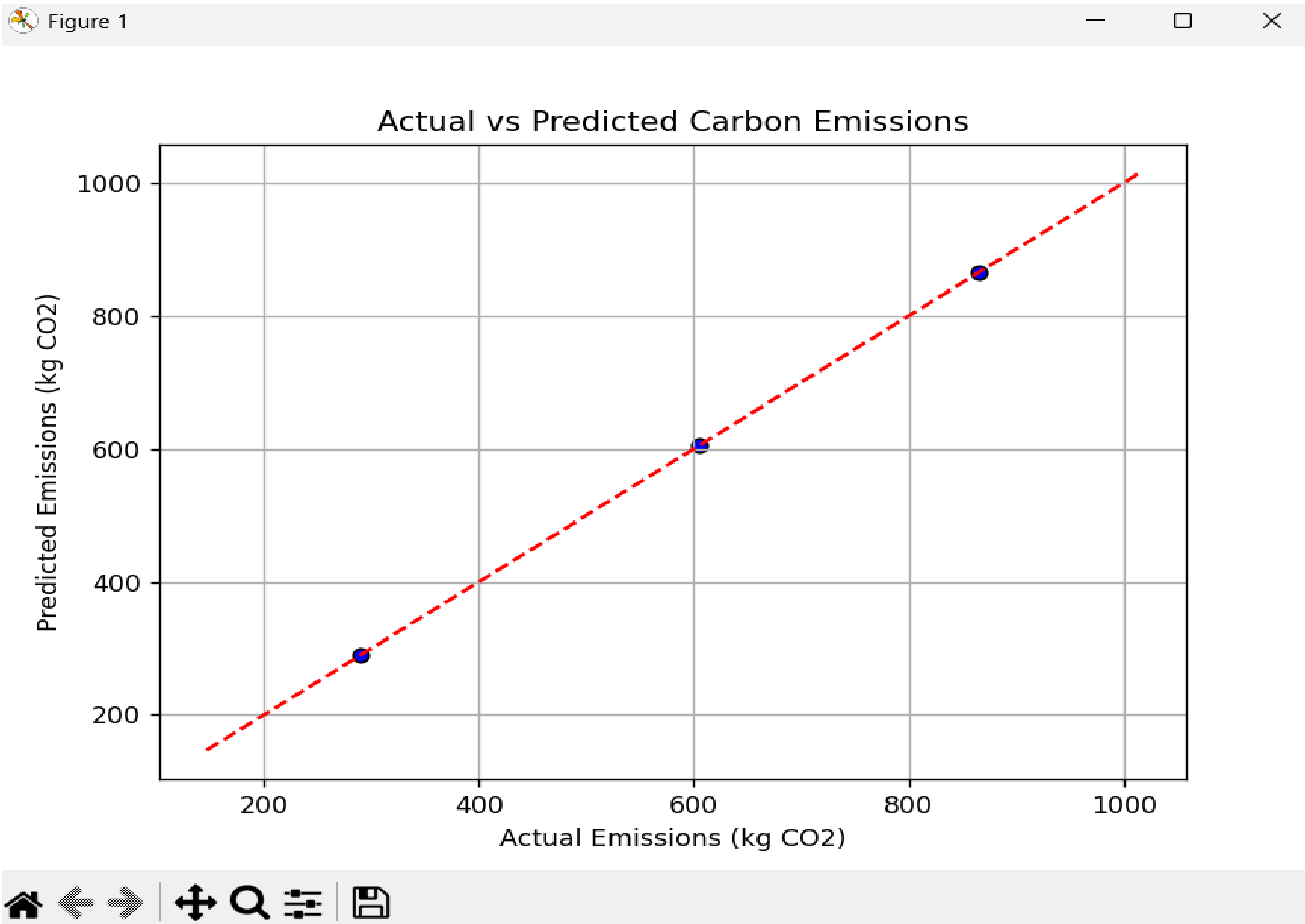
```
    'Distance_km': [15, 30, 50, 60, 25, 80, 55, 100, 110, 130],
}

df = pd.DataFrame(data)

# Step 3: Calculate carbon emissions using emission factors
# (Average emission factors)
EF_electricity = 0.82   # kg CO2 per kWh
EF_fuel = 2.31          # kg CO2 per litre
EF_water = 0.0003       # kg CO2 per litre
EF_distance = 0.12      # kg CO2 per km

df['Carbon_Emission(kgCO2)'] = (
    df['Electricity_kWh'] * EF_electricity +
    df['Fuel_Litres'] * EF_fuel +
    df['Water_Litres'] * EF_water +
    df['Distance_km'] * EF_distance
)

print("\n=== Dataset with Carbon Emissions ===")
print(df)
```

```python

# Step 4: Split dataset for ML
X = df[['Electricity_kWh', 'Fuel_Litres', 'Water_Litres', 'Distance_km']]
y = df['Carbon_Emission(kgCO2)']

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=42)

# Step 5: Train Linear Regression model
model = LinearRegression()
model.fit(X_train, y_train)

# Step 6: Predict emissions
y_pred = model.predict(X_test)

# Step 7: Evaluate model performance
mae = mean_absolute_error(y_test, y_pred)
r2 = r2_score(y_test, y_pred)

print("\n=== Model Performance ===")
print(f"Mean Absolute Error: {mae:.3f}")
print(f"R² Score: {r2:.3f}")
```

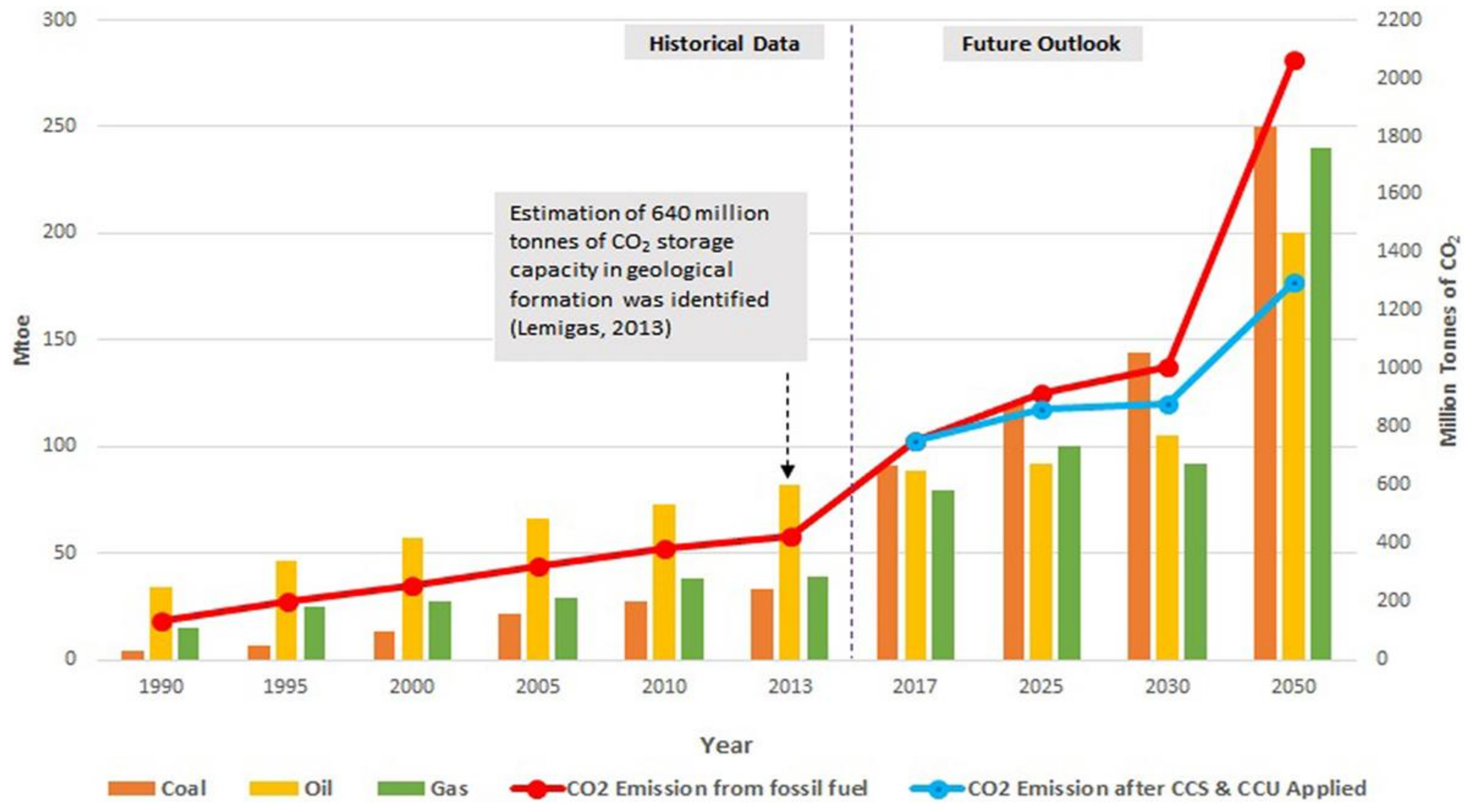
```python

# Step 8: Visualize Actual vs Predicted emissions
plt.figure(figsize=(7,5))
plt.scatter(y_test, y_pred, color='blue', edgecolors='black')
plt.plot([y.min(), y.max()], [y.min(), y.max()], color='red', linestyle='--')
plt.title('Actual vs Predicted Carbon Emissions')
plt.xlabel('Actual Emissions (kg CO2)')
plt.ylabel('Predicted Emissions (kg CO2)')
plt.grid(True)
plt.show()

# Step 9: Predict for new input (future data)
print("\n=== Future Prediction Example ===")
new_data = pd.DataFrame({
    'Electricity_kWh': [600],
    'Fuel_Litres': [75],
    'Water_Litres': [2500],
    'Distance_km': [90]
})

predicted_emission = model.predict(new_data)
print(f"Predicted Carbon Emission: {predicted_emission[0]:.2f} kg CO₂")
```

```python
predicted_emission = model.predict(new_data)
print(f"Predicted Carbon Emission: {predicted_emission[0]:.2f} kg CO₂")

# Step 10: Save model (optional)
import joblib
joblib.dump(model, 'carbon_footprint_model.pkl')
print("\nModel saved as carbon_footprint_model.pkl ✅")

```

Results and Comparision



```
PS C:\Users\pradu\python_chapter 3> python -u "c:\Users\pradu\python_chapter 3\CARBOON.PY"

=== Dataset with Carbon Emissions ===
   Electricity_kWh  Fuel_Litres  Water_Litres  Distance_km  Carbon_Emission(kgCO2)
0              120           20           800           15                  146.64
1              250           35          1200           30                  289.81
2              310           50          1500           50                  376.15
3              400           65          2000           60                  485.95
4              180           25          1000           25                  208.65
5              500           80          2500           80                  605.15
6              320           55          1700           55                  396.56
7              610          100          3000          100                  744.10
8              700          120          3300          110                  865.39
9              850          130          3500          130                 1013.95

=== Model Performance ===
Mean Absolute Error: 0.000
R² Score: 1.000

=== Future Prediction Example ===
Predicted Carbon Emission: 676.80 kg CO₂

Model saved as carbon_footprint_model.pkl ✅
PS C:\Users\pradu\python_chapter 3>
```


Historical Data
Future Outlook
Estimation of 640 million tonnes of CO_2 storage capacity in geological formation was identified (Lemigas, 2013)
Mtoe
Million Tonnes of CO_2
300
250
200
150
100
50
0
2200
2000
1800
1600
1400
1200
1000
800
600
400
200
0
1990
1995
2000
2005
2010
2013
2017
2025
2030
2050
Year
Coal
Oil
Gas
CO2 Emission from fossil fuel
CO2 Emission after CCS & CCU Applied

Conclusion

Summary of Key Contributions

The integration of Machine Learning (ML) and Artificial Intelligence (AI) into carbon footprint tracing represents a transformative approach to environmental sustainability. By harnessing advanced algorithms and real-time data analytics, organizations can achieve unprecedented accuracy and granularity in measuring and managing their carbon emissions across Scope 1, Scope 2, and Scope 3 categories.

AI-driven systems facilitate the automation of data collection, processing, and analysis, enabling dynamic monitoring and reporting of emissions. These technologies support proactive decision-making by identifying emission hotspots, forecasting future trends, and evaluating the effectiveness of mitigation strategies. Moreover, AI's capability to process complex datasets, including unstructured information from supply chains and operational activities, enhances the comprehensiveness of carbon accounting. However, it is crucial to acknowledge the environmental impact of AI technologies themselves. The energy consumption associated with training and deploying AI models can contribute to carbon emissions, particularly when powered by non-renewable energy sources. Therefore, adopting energy-efficient AI practices and utilizing renewable energy are essential to ensure that the deployment of AI in carbon footprint tracing aligns with sustainability objectives.

In conclusion, while AI and ML offer powerful tools for enhancing carbon footprint tracing, their implementation must be accompanied by strategies to mitigate their own environmental impact. Through responsible deployment and continuous optimization, AI can significantly contribute to global efforts in reducing carbon emissions and combating climate change.

Future Work

1. Integration with IoT and Smart Sensors
In the future, carbon footprint tracing systems can be integrated with Internet of Things (IoT) devices to collect real-time data on energy usage, fuel consumption, temperature, and emissions. Smart meters and sensors will automate the data-gathering process, improving accuracy and eliminating manual input errors.

2. Development of Real-Time Monitoring Dashboards
Advanced AI-based dashboards can be built for live tracking and visualization of carbon emissions across different sectors like industry, transport, and households. These dashboards will provide instant feedback and alerts when emission levels exceed set thresholds.

3. Enhanced Predictive Modeling with Deep Learning
Future research can focus on using deep learning models such as LSTM (Long Short-Term Memory) and GRU (Gated Recurrent Unit) for time-series forecasting of emissions. These models will help predict future emission trends with higher accuracy based on historical data and dynamic variables.

Incorporation of Satellite and GIS Data
Integrating satellite imagery and Geographical Information Systems (GIS) will allow region-based emission mapping and spatial analysis. This can help governments and organizations monitor large-scale pollution sources and track the carbon footprint of urban and rural regions.

5. Personalized Carbon Footprint Tracking Applications
AI can be used to develop mobile or web-based personal assistants that help individuals monitor their daily carbon footprint based on activities like transportation, diet, and energy use. These apps can suggest personalized ways to reduce carbon emissions using recommender systems.

6. Integration with Blockchain for Transparency
Combining AI with Blockchain technology can ensure transparent and tamper-proof recording of carbon emission data. This will be particularly useful for carbon credit trading systems and for maintaining trust in sustainability reports.

7. AI-Driven Policy Recommendations
Machine Learning models can be trained on global climate datasets to assist policymakers in formulating emission reduction strategies. AI can identify the most effective actions to achieve net-zero targets for industries or cities.

8. Automation of Carbon Offset Programs
Future systems may automatically suggest or execute carbon offset actions—such as tree plantation drives or renewable energy investments—based on users' emission levels. Reinforcement learning algorithms can optimize these offset decisions for maximum impact.

9. Integration with Renewable Energy Systems
AI-powered carbon tracking can be linked with renewable energy management systems, such as solar or wind networks, to automatically adjust energy usage and minimize emissions based on real-time conditions.

References

- **Awan, U., Nauman, S., & Sroufe, R. (2023).** *Artificial Intelligence for Sustainable Carbon Management: Applications and Challenges.* Journal of Cleaner Production, 398, 136749.

- **Zhao, X., Wang, S., & Li, D. (2022).** *Machine Learning-Based Carbon Emission Prediction Model for Smart Cities.* IEEE Access, 10, 125611–125624.

- **Kumar, A., & Singh, R. (2022).** *AI-Driven Approach for Carbon Footprint Estimation in Industrial Systems.* International Journal of Environmental Science and Technology, 19(7), 6853–6866.

- **Mohan, S., & Bhatia, M. (2021).** *Carbon Footprint Analytics Using Machine Learning: A Predictive Framework.* Environmental Modelling & Software, 142, 105102.

- **International Energy Agency (IEA). (2023).** *CO_2 Emissions Data and Analysis.*

- **Zhang, L., Chen, H., & Xu, J. (2021).** *Deep Learning-Based Prediction of Carbon Emissions for Energy Optimization.* Energy Reports, 7, 5871–5883.

- **García, J., & Torres, P. (2020).** *Data-Driven Methods for Carbon Emission Forecasting in Transportation Using AI Models.* Sustainable Computing: Informatics and Systems, 28, 100446.

- **World Bank Group. (2023).** *Carbon Footprint Measurement and Reduction Strategies: A Data Perspective.*

- **yadav, V., & Sharma, P. (2024).** *Integrating IoT, AI, and ML for Real-Time Carbon Footprint Tracking in Smart Environments.* IEEE Internet of Things Journal, 11(2), 1802–1814.

- **Mohan, S., & Bhatia, M. (2021).** Carbon Footprint Analytics Using Machine Learning: A Predictive Framework. *Environmental Modelling & Software, 142*, 105102.

- **García, J., & Torres, P. (2020).** Data-Driven Methods for Carbon Emission Forecasting in Transportation Using AI Models. *Sustainable Computing: Informatics and Systems, 28*, 100446**.**

- **Chen, Z. (2023).** Prediction of Carbon Dioxide Emissions Based on Machine Learning Algorithms. *Applied and Computational Engineering, 15*(10), 102–115.

- **Liu, X., Yu, H., Liu, H., & Sun, Z. (2025).** Multi-Factor Carbon Emissions Prediction in Coal-Fired Power Plants: A Machine Learning Approach for Carbon Footprint Management. *Energies, 18*(7), 1715.

- **Liu, X., Yu, H., Liu, H., & Sun, Z. (2025).** Multi-Factor Carbon Emissions Prediction in Coal-Fired Power Plants: A Machine Learning Approach for Carbon Footprint Management. *Energies, 18*(7), 1715

- **Lu, Y. (2024).** A Review of Machine Learning Used in Carbon Emission Prediction. *Advances in Engineering Technology Research, 12*(1), 2942–2955.