ENHANCED VEHICLE CO2 EMISSION PREDICTION USING LSTM NETWORKS AND OBD-II DATA

Ch. Vinay Kumar[1], Kola Jhansi[2] Matti Nithin[3], Mohd Naseer Baba[4], Mohd Habeeb Ur Rahamn[5]

[1] *Assistant Professor*, Department *of CSE (ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING) TKR COLLEGE OF ENGINEERING & TECHNOLOGY*

[2,3,4,5] *UG Scholars in Department of CSE (ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING) TKR COLLEGE OF ENGINEERING & TECHNOLOG*

ABSTRACT: In recent years, the rise in carbon dioxide (CO2) emissions from vehicles has become a critical concern for environmental sustainability and public health. Direct measurement of emissions from every vehicle is costly and impractical; however, modern vehicles are equipped with On-Board Diagnostics (OBD-II) systems that provide real-time telematics data. This project utilizes such data—including engine RPM, vehicle speed, throttle position, and fuel flow—to estimate CO2 emissions using a Long Short-Term Memory (LSTM) neural network. LSTM models are well-suited for time-series analysis, enabling the system to capture temporal patterns in vehicle behavior. The proposed model is trained on sensor data to predict CO2 emissions with high accuracy over time. The results demonstrate that LSTM-based prediction can effectively estimate emissions without the need for expensive physical measurement systems. This approach provides a scalable, cost-effective, and real-time solution for emission monitoring, supporting smart transportation systems, pollution control strategies, and environmentally sustainable policies.

Keywords: CO2 Emissions, LSTM, OBD-II, Time-Series Data, Vehicle Telematics, Emission Prediction, Deep Learning

1. INTRODUCTION

Climate change is one of the most serious global threats today, driven largely by greenhouse gases such as carbon dioxide (CO_2), methane (CH_4), and nitrous oxide (N_2O). Since the Industrial Revolution, the average global temperature has increased by approximately 1°C, leading to more extreme weather and environment disruptions. In India, about 23.6% of total CO_2 emissions come from the transport sector. Of this, a significant 65.4% is attributed to road transport, including cars, trucks, and other vehicles on highways and city roads. Monitoring emissions from all these vehicles is a major challenge. Installing CO_2 sensors in every vehicle is expensive and impractical. Therefore, there is a strong need for a scalable, real-time solution that can monitor vehicle emissions effectively without adding extra hardware.

To overcome the limitations of physical sensors, our approach uses data already available in most vehicles through the OBD-II (On-Board Diagnostics) port. This data can be sent to the cloud using IoT dongles, where it is analyzed to predict CO_2

emissions. We propose using an LSTM (Long Short-Term Memory) deep learning model, which is especially suited for time-series data like vehicle sensor readings. It learns from past patterns and relationships over time, making it more effective than traditional machine learning models that treat each data point independently.

The model uses key vehicle parameters such as RPM, speed, engine load, throttle position, fuel flow, and manifold pressure to estimate emissions. LSTM is robust in handling noisy, real-world data and captures the sequential nature of driving behavior. This approach enables real-time CO_2 monitoring without the need for new sensors. It's cost effective, scalable, and can be applied to thousands of vehicles with minimal infrastructure.

1.1 Motivation

The rapid growth in the number of vehicles has resulted in a significant rise in CO_2 emissions, posing serious threats to the environment and contributing to global climate change. Monitoring and reducing vehicular emissions have therefore become essential steps toward achieving sustainable transportation. There is a growing need for intelligent systems that can analyze vehicle data in real time, raise environmental awareness, and encourage drivers to adopt eco-friendly driving habits. By leveraging telematics data and deep learning techniques, such systems can help users understand their vehicle's emission patterns and actively contribute to reducing their carbon footprint, thereby supporting global sustainability goals.

Proposed System:

The proposed system is designed to make vehicle CO_2 emission monitoring smarter and more effective while increasing environmental awareness among drivers. It collects real-time data from the vehicle's OBD-II port, which includes important details like engine speed, throttle position, fuel flow, and engine load. By using this data, the system can better understand how a vehicle performs and how much CO_2 it produces while running.

This information is processed using a deep learning model that studies time-based data to predict CO_2 emission levels accurately. The system keeps track of emission trends and detects any changes that may affect a vehicle's performance or increase emissions. Through this continuous monitoring and analysis, it provides meaningful insights that help drivers and vehicle owners better understand their vehicle's impact on the environment.

The system also encourages users to adopt eco-friendly driving habits by showing how their driving style affects emission levels. By driving more smoothly and maintaining their vehicles properly, users can help reduce CO_2 emissions and

2. Use Case Diagram

A use case diagram is one of the simplest and most effective ways to represent how a user interacts with a system. It visually illustrates the functional requirements of a system by showing the different actors and the various use cases through which they interact with the system. Use case diagrams are typically created alongside textual use case descriptions and are often complemented by other UML diagrams. They help stakeholders understand the system's functionality at a high level, making it easier to identify user needs, clarify requirements, and design the overall system flow.

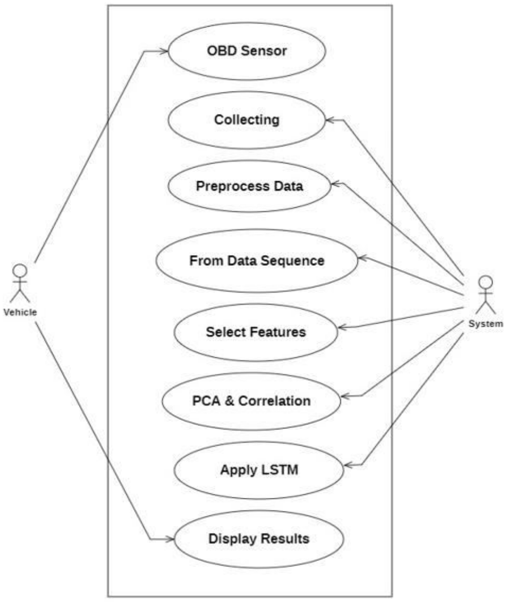


Fig.:1 use case diagram

Sequence Diagram

Sequence Diagram

A sequence diagram in Unified Modelling Language (UML) is a type of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. A sequence diagram shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects required to carry out the functionality of the scenario. Sequence diagrams are typically associated with use case realizations in the Logical View of the system under development.

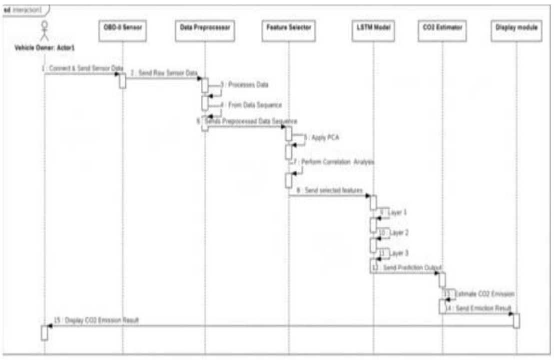


Fig.2: Sequence Diagram

3. Algorithms

Long Short-Term Memory (LSTM)

LSTM is a type of Recurrent Neural Network (RNN) designed to process sequential and time-series data. It uses memory cells and gating mechanisms to store and update information over time, allowing it to learn long-term dependencies in data.In this system, vehicle parameters such as RPM, speed, throttle position, and engine load change continuously over time. LSTM captures these temporal relationships and learns patterns in vehicle behavior. This helps in accurately predicting CO_2 emissions based on past and current data, making the system reliable for time-based prediction.

Min-Max Normalization

Min-Max Normalization is a preprocessing technique that scales input data into a fixed range, usually between 0 and 1. It ensures that all features have equal importance during model training. Vehicle parameters have different ranges (e.g., RPM is high, throttle is low). Normalization brings all values to a common scale, improving model training speed and accuracy. It prevents large values from dominating the learning process.

4Sliding Window Technique

The sliding window technique is used to convert continuous time-series data into smaller sequences of fixed size. These sequences are then used as input to the model. Instead of using single data points, the system uses a group of consecutive values to understand short-term trends. This helps the LSTM model learn patterns more effectively and improves prediction accuracy.

Correlation Analysis

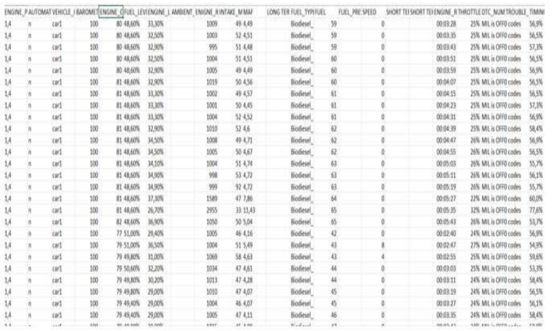
Correlation Analysis is a statistical technique used to measure the relationship between two or more variables. It determines how strongly variables are related and whether the relationship is positive, negative, or neutral. The correlation value typically ranges from -1 to +1, where +1 indicates a strong positive relationship, -1 indicates a strong negative relationship, and 0 indicates no relationship. In this system, correlation analysis is used to understand the relationship between vehicle parameters such as RPM, speed, throttle position, and engine load. It helps identify which features have the most significant impact on CO_2 emissions. By analyzing these relationships using a heatmap, the system provides better insights into how different parameters influence emission levels, improving model interpretation and analysis.

Principal Component Analysis (PCA)

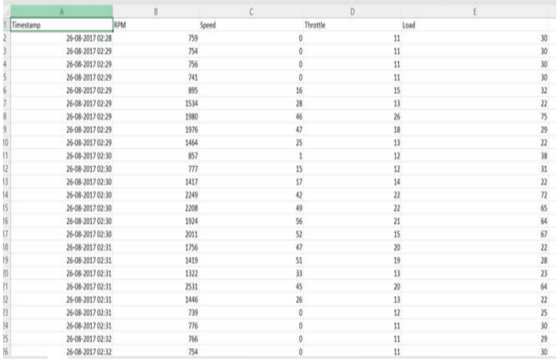
Principal Component Analysis (PCA) is a dimensionality reduction technique used to transform a large set of variables into a smaller set of uncorrelated variables called principal components. It reduces redundancy in data while retaining the most important information. In this project, PCA can be used to reduce the number of input features while preserving essential information. This helps in improving computational efficiency and reducing model complexity. By removing redundant or highly correlated features, PCA can enhance the performance of the LSTM model and make the system faster and more efficient.

Sample data

The sample data used in this project consists of vehicle telematics data collected from On-Board Diagnostics (OBD-II) systems. This data is used to train and test the deep learning model for predicting carbon dioxide (CO_2) emissions based on vehicle operating conditions. The dataset is stored in structured formats such as CSV files, containing time-series sensor readings recorded at regular intervals. The data represents real-world driving conditions and includes parameters such as engine RPM, vehicle speed, throttle position, fuel consumption, and engine load. These features capture the dynamic behavior of the vehicle under different operating scenarios, including acceleration, cruising, and idling conditions. Each record in the dataset corresponds to a specific timestamp, representing the state of the vehicle at that moment. The dataset is preprocessed and organized into input sequences using a sliding window approach, which enables the LSTM model to learn temporal patterns in the data. The corresponding output is the predicted CO_2 emission value for each sequence. The dataset is divided into training and testing sets. The training data is used to train the LSTM model, while the testing data is used to evaluate its performance. A sample input consists of a sequence of sensor readings, and the output is the predicted CO_2 emission value. The sample data plays a crucial role in determining the accuracy and reliability of the emission prediction.



ENGINE_P	AUTOMAT	VEHICLE_I	BAROMET	ENGINE_C	FUEL_LEV	ENGINE_L	AMBIENT_	ENIGNE_R	INTAKE_M	MAF	LONG TER	FUEL_TYP	FUEL	FUEL_PRE	SPEED	SHORT TE	SHORT TE	ENGINE_R	THROTTLE	DTC_NUM	TROUBLE_	TIMIN
1,4	n	car1	100	80	48,60%	33,30%		1009	49	4,49		Biodiesel_	59		0			00:03:28	25%	MIL is OFF0 codes		56,9%
1,4	n	car1	100	80	48,60%	32,50%		1003	52	4,51		Biodiesel_	59		0			00:03:35	25%	MIL is OFF0 codes		56,5%
1,4	n	car1	100	80	48,60%	32,90%		995	51	4,48		Biodiesel_	59		0			00:03:43	25%	MIL is OFF0 codes		57,3%
1,4	n	car1	100	80	48,60%	32,50%		1004	51	4,51		Biodiesel_	60		0			00:03:51	25%	MIL is OFF0 codes		56,5%
1,4	n	car1	100	80	48,60%	32,90%		1005	49	4,49		Biodiesel_	60		0			00:03:59	25%	MIL is OFF0 codes		56,9%
1,4	n	car1	100	81	48,60%	32,90%		1019	50	4,56		Biodiesel_	60		0			00:04:07	25%	MIL is OFF0 codes		56,5%
1,4	n	car1	100	81	48,60%	33,30%		1002	49	4,57		Biodiesel_	61		0			00:04:15	25%	MIL is OFF0 codes		56,5%
1,4	n	car1	100	81	48,60%	33,30%		1001	50	4,45		Biodiesel_	61		0			00:04:23	25%	MIL is OFF0 codes		57,3%
1,4	n	car1	100	81	48,60%	33,30%		1004	52	4,52		Biodiesel_	61		0			00:04:31	25%	MIL is OFF0 codes		56,9%
1,4	n	car1	100	81	48,60%	32,90%		1010	52	4,6		Biodiesel_	62		0			00:04:39	25%	MIL is OFF0 codes		58,4%
1,4	n	car1	100	81	48,60%	34,50%		1008	49	4,71		Biodiesel_	62		0			00:04:47	26%	MIL is OFF0 codes		56,9%
1,4	n	car1	100	81	48,60%	34,50%		1005	50	4,67		Biodiesel_	62		0			00:04:55	26%	MIL is OFF0 codes		56,5%
1,4	n	car1	100	81	48,60%	34,10%		1004	51	4,74		Biodiesel_	63		0			00:05:03	26%	MIL is OFF0 codes		55,7%
1,4	n	car1	100	81	48,60%	34,90%		998	53	4,72		Biodiesel_	63		0			00:05:11	26%	MIL is OFF0 codes		56,1%
1,4	n	car1	100	81	48,60%	34,90%		999	92	4,72		Biodiesel_	63		0			00:05:19	26%	MIL is OFF0 codes		55,7%
1,4	n	car1	100	81	48,60%	37,30%		1589	47	7,86		Biodiesel_	64		0			00:05:27	22%	MIL is OFF0 codes		60,0%
1,4	n	car1	100	81	48,60%	26,70%		2955	33	11,43		Biodiesel_	65		0			00:05:35	32%	MIL is OFF0 codes		77,6%
1,4	n	car1	100	82	48,60%	36,90%		1050	50	5,04		Biodiesel_	65		0			00:05:43	26%	MIL is OFF0 codes		53,7%
1,4	n	car1	100	77	51,00%	29,40%		1005	46	4,16		Biodiesel_	42		0			00:02:40	24%	MIL is OFF0 codes		56,9%
1,4	n	car1	100	79	51,00%	36,50%		1004	51	5,49		Biodiesel_	43		8			00:02:47	27%	MIL is OFF0 codes		54,9%
1,4	n	car1	100	79	49,80%	31,00%		1069	58	4,63		Biodiesel_	43		4			00:02:55	25%	MIL is OFF0 codes		59,6%
1,4	n	car1	100	79	50,60%	32,20%		1034	47	4,61		Biodiesel_	44		0			00:03:03	25%	MIL is OFF0 codes		53,3%
1,4	n	car1	100	79	49,80%	30,20%		1013	47	4,28		Biodiesel_	44		0			00:03:11	24%	MIL is OFF0 codes		58,4%
1,4	n	car1	100	79	49,80%	29,00%		1010	47	4,07		Biodiesel_	45		0			00:03:19	24%	MIL is OFF0 codes		56,5%
1,4	n	car1	100	79	49,40%	29,00%		1004	46	4,07		Biodiesel_	45		0			00:03:27	24%	MIL is OFF0 codes		56,1%
1,4	n	car1	100	79	49,40%	29,00%		1005	47	4,11		Biodiesel_	46		0			00:03:35	24%	MIL is OFF0 codes		58,4%



	A	B	C	D	E
1	Timestamp	RPM	Speed	Throttle	Load
2	26-08-2017 02:28	759	0	11	30
3	26-08-2017 02:29	754	0	11	30
4	26-08-2017 02:29	756	0	11	30
5	26-08-2017 02:29	741	0	11	30
6	26-08-2017 02:29	895	16	15	32
7	26-08-2017 02:29	1534	28	13	22
8	26-08-2017 02:29	1980	46	26	75
9	26-08-2017 02:29	1976	47	18	29
10	26-08-2017 02:29	1464	25	13	22
11	26-08-2017 02:30	857	1	12	38
12	26-08-2017 02:30	777	15	12	31
13	26-08-2017 02:30	1417	17	14	22
14	26-08-2017 02:30	2249	42	22	72
15	26-08-2017 02:30	2208	49	22	65
16	26-08-2017 02:30	1924	56	21	64
17	26-08-2017 02:30	2011	52	15	67
18	26-08-2017 02:31	1756	47	20	22
19	26-08-2017 02:31	1419	51	19	28
20	26-08-2017 02:31	1322	33	13	23
21	26-08-2017 02:31	2531	45	20	64
22	26-08-2017 02:31	1446	26	13	22
23	26-08-2017 02:31	739	0	12	25
24	26-08-2017 02:31	776	0	11	30
25	26-08-2017 02:32	766	0	11	29
26	26-08-2017 02:32	754	0	11	30

4. IMPLEMENTATION & RESULTS

This chapter explains the implementation and results of the CO_2 Emission Prediction System, highlighting its modular functions and output screens. The system integrates data preprocessing, time-series modeling using LSTM, real-time data simulation, prediction, and visualization into a unified pipeline. The implementation is carried out using Python with Tkinter for GUI, TensorFlow/Keras for deep learning, and Matplotlib/Seaborn for visualization. The output screens include the main interface, real-time monitoring dashboard, prediction graphs, and statistical analysis window. The results demonstrate accurate CO_2 emission prediction, stable outputs, and effective visualization, confirming the system's ability to provide real-time emission monitoring.

4.1 Explanation of Key Functions

Data Loading Function

The system loads vehicle telematics data from CSV files.
The dataset contains parameters such as RPM, speed, throttle position, and engine load. The data is read using Pandas and prepared for further processing.

Data Preprocessing Function

The input data is preprocessed before training.
This includes:

- Normalization using Min-Max scaling
- Conversion into time-series sequences using a sliding window This step helps the LSTM model learn temporal dependencies effectively.

Model Training Function

The system uses a Long Short-Term Memory (LSTM) neural network.

- Input: Time-series sequences
- Output: CO_2 emission values The model is trained using multiple epochs and optimized using mean squared error loss.

Prediction Function

The trained model is used to predict CO_2 emissions.
Input sequences are passed to the model, and predicted values are converted back to

original scale. The results are displayed graphically and in textual format.

Real-Time Simulation Function

The system supports real-time data simulation.
Vehicle parameters such as RPM, speed, throttle, and load are generated dynamically or received via Bluetooth. The GUI updates continuously with live values.

Visualization Function

The system visualizes results using graphs.

- Live RPM graph
- CO_2 prediction graph These visualizations help in understanding trends and patterns.

Statistical Analysis Function

The system performs statistical analysis of predictions.
It generates:

- Correlation heatmap
- Boxplot for CO_2 distribution This helps in analyzing relationships between features and emissions.

User Interface Function

The system includes a GUI developed using Tkinter.
Users can:

- Upload dataset
- Start/stop simulation
- View predictions
- Perform analysis

Method of Implementation

The CO_2 Emission Prediction System is implemented using a modular approach combining a graphical user interface, data processing, and a deep learning model. The system is developed using Python, where Tkinter is used for the GUI, Pandas and NumPy for data handling, Scikit-learn for preprocessing, TensorFlow/Keras for building the LSTM model, and Matplotlib/Seaborn for visualization.

Initially, the user uploads a dataset or connects to a Bluetooth device. The data is preprocessed using normalization and converted into time-series sequences using a sliding window technique. The LSTM model is then trained on the processed data to learn temporal patterns. During execution, the system supports real-time data simulation or playback of recorded data. The trained model predicts CO_2 emissions, and the results are displayed through graphs and output panels. Additionally, statistical analysis is performed to evaluate relationships between features and predictions.

Forms

The system consists of a graphical user interface (GUI) developed using Tkinter, which serves as the main form for user interaction. The form provides all functionalities within a single window, making the system simple and user-friendly.

The form includes various components such as:

- Buttons for uploading CSV files and connecting to a Bluetooth device
- Control buttons such as Start, Stop, and Reset

- Prediction button to display CO₂ emissions
- Statistical Analysis button to view advanced results

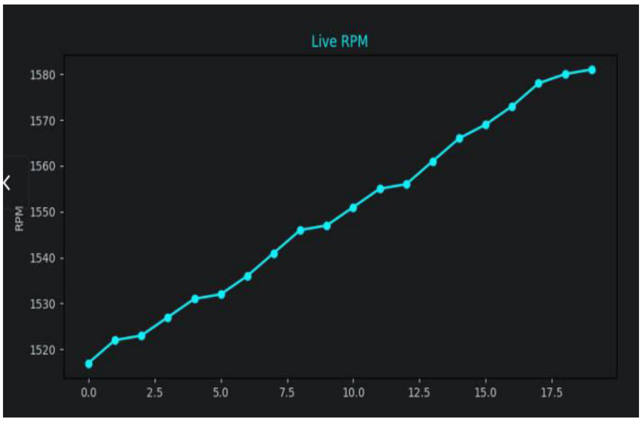
The interface also displays real-time vehicle parameters such as timestamp, RPM, speed, throttle position, and engine load. Graphical components like the live RPM chart and CO₂ prediction graph are embedded within the form to provide visual insights.

The form acts as the central control panel of the system, enabling efficient interaction between the user and the application.

Fig 4.1 Main User Interface Form of CO₂ Emission Prediction System

Output Screens

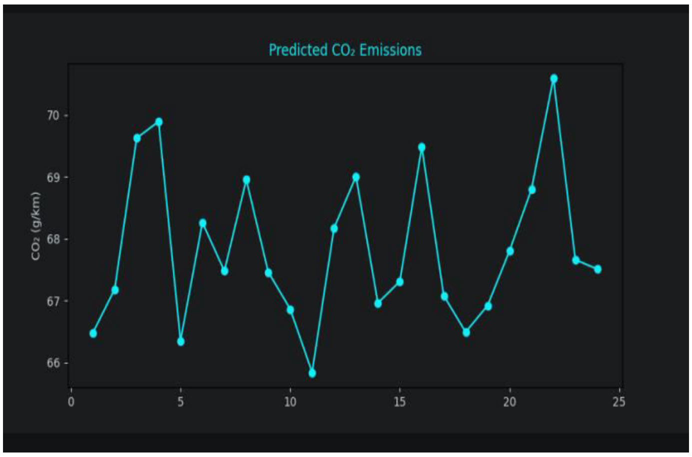
The system generates multiple output screens to visualize real-time data, prediction results, and statistical analysis. These outputs help in understanding the behavior of the model and evaluating the performance of the CO₂ emission prediction system.



Live RPM Graph

Fig Live RPM Graph

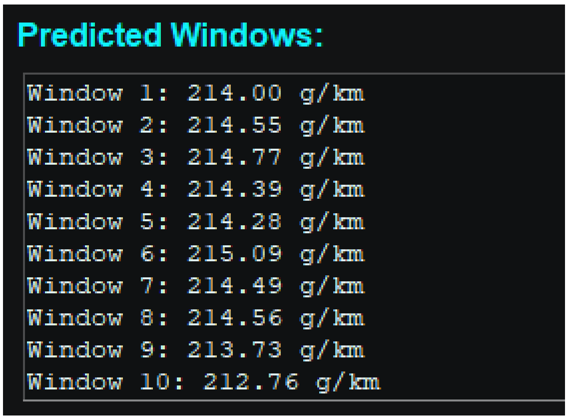
The Live RPM Graph displays the real-time variation of engine RPM values during system execution. It continuously updates as new data is processed, allowing users to monitor engine performance dynamically. This visualization helps in understanding how RPM changes over time under different operating conditions.



CO₂ Emission Prediction Graph

: CO₂ Emission Prediction Graph

This graph illustrates the predicted CO₂ emission values generated by the LSTM model over time. It shows the trend and variation of emissions based on input features such as RPM, speed, throttle position, and engine load. The graph provides a clear visual representation of how emissions fluctuate during vehicle operation.



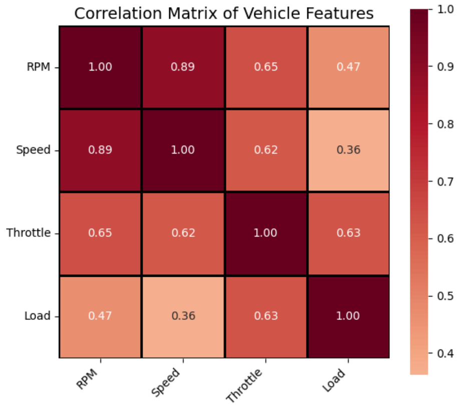
```
Predicted Windows:

Window 1: 214.00 g/km
Window 2: 214.55 g/km
Window 3: 214.77 g/km
Window 4: 214.39 g/km
Window 5: 214.28 g/km
Window 6: 215.09 g/km
Window 7: 214.49 g/km
Window 8: 214.56 g/km
Window 9: 213.73 g/km
Window 10: 212.76 g/km
```

Predicted CO_2 Output Values

Predicted CO_2 Output Values

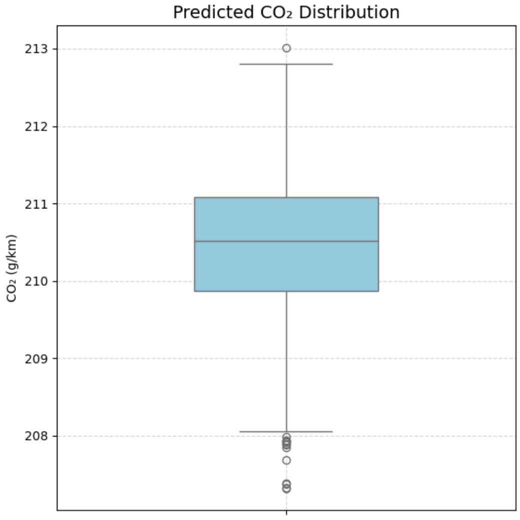
The output panel displays the predicted CO_2 emission values in textual format for each time window. Each entry represents the emission value corresponding to a specific sequence of input data. This detailed numerical output helps in analyzing the exact prediction values generated by the model.



Correlation Heatmap

Correlation Heatmap

The correlation heatmap represents the relationship between different vehicle parameters such as RPM, speed, throttle position, and engine load. It uses color intensity to indicate the strength of correlation between variables. This visualization helps in identifying which features have a stronger influence on CO_2 emissions.



CO_2 Emission Boxplot

CO_2 Emission Boxplot

The boxplot shows the statistical distribution of predicted CO_2 emission values. It highlights important metrics such as median, quartiles, and outliers. This helps in understanding the variability and consistency of the predictions made by the model.

Result Analysis

The results obtained from the system demonstrate the effectiveness of the LSTM model in predicting CO_2 emissions using vehicle telematics data. The predicted emission values follow logical patterns based on input features such as RPM, speed, throttle position, and engine load.

The CO_2 prediction graph shows smooth variations over time, indicating that the model successfully captures temporal dependencies in the data. The live RPM graph reflects real-time engine behavior, which directly influences emission levels.

The correlation heatmap reveals strong relationships between parameters such as RPM and speed, indicating their significant impact on CO_2 emissions. The boxplot analysis shows that the predicted values are consistently distributed with minimal outliers, indicating stable model performance.

Overall, the system provides accurate and reliable predictions with efficient real-time processing. The results confirm that the proposed approach is effective for monitoring and analyzing vehicle emissions.

5. CONCLUSION

The present work successfully demonstrates the design and implementation of an enhanced vehicle CO_2 emission prediction system using Long Short-Term Memory (LSTM) networks and OBD-II telematics data. The system provides an efficient, cost-effective, and real-time solution for estimating vehicle emissions without the need for expensive physical measurement equipment. By utilizing readily available sensor data such as engine RPM, vehicle speed, throttle position, and engine load, the system effectively models the relationship between driving behavior and CO_2 emissions.

The proposed system integrates multiple components, including data preprocessing, time-series sequence generation, LSTM-based prediction, real-time simulation, and graphical visualization, into a unified framework. The use of LSTM enables the model to capture temporal dependencies in sequential data, resulting in accurate and stable emission predictions. The implementation of normalization and sliding window techniques further improves the learning capability of the model and ensures consistency in predictions.

The graphical user interface developed using Tkinter enhances usability by providing real-time data display, live RPM monitoring, emission prediction graphs, and statistical analysis tools such as correlation heatmaps and boxplots. The system supports both real-time simulation and dataset playback, making it flexible for different use cases. The statistical analysis module provides deeper insights into the relationships between input parameters and predicted emissions, aiding in better understanding of vehicle behavior.

Overall, the system achieves its objective of delivering a scalable, reliable, and real-time CO_2 emission prediction solution. It demonstrates that combining deep learning techniques with vehicle telematics data can significantly contribute to environmental monitoring and smart transportation systems. This project highlights the potential of data-driven approaches in supporting pollution control strategies and promoting sustainable and eco-friendly transportation practices.

6. REFERENCE

[1] M. Singh and R. K. Dubey, “Deep learning model based c02 emissions prediction using vehicles telematics sensors data,” IEEE Transactions on Intelligent Vehicles, vol. 8, no.1, 2023.

[2] R. Zhang, Y. Wang, Y. Pang, B. Zhang, Y. Wei, M. Wang, and R. Zhu, “A deep learning micro-scale model to estimate the

CO_2 emissions from light-duty diesel trucks based on real-world driving," Atmosphere, vol. 13, no. 9, p. 1466, 2022.

[3] S. Kumari and S. K. Singh, "Machine learning-based time series models for effective CO_2 emission prediction in India," Environmental Science and Pollution Research, vol. 30, pp. 116601–116616, 2023.

[4] T. Jia, P. Zhang, and B. Chen, "A microscopic model of vehicle CO_2 emissions based on deep learning — a spatiotemporal analysis of taxicabs in Wuhan, China," IEEE Transactions on Intelligent Transportation Systems, vol. 23, no. 10, pp. 18446–18455, 2022.

[5] Y. Meng and H. Noman, "Predicting CO_2 emission footprint using AI through machine learning," Atmosphere, vol. 13, no. 11, p. 1871, 2022.

[6] N. Niroomand, C. Bach, and M. Elser, "Segment-based CO_2 emission evaluations from passenger cars based on deep learning techniques," *IEEE Access*, vol. 9, pp. 166314–166327, 2021.

[7] H.-T. Wen, J.-H. Lu, and D.-S. Jhang, "Features importance analysis of diesel vehicles' NO_x and CO_2 emission predictions in real road driving based on gradient boosting regression model," *International Journal of Environmental Research and Public Health*, vol. 18, no. 24, Art. 13044, 2021.

[8] M. Rahman, Y. Zhou, J. Rogers, V. Chen, M. Sattler, and K. Hyun, "A comparative assessment of CO_2 emission between gasoline, electric, and hybrid vehicles: a well-to-wheel perspective using agent-based modeling," Journal of Cleaner Production, vol. 321, Art. 128931, 2021.

[9] R. Shang, Y. Zhang, and Z.-J. Max Shen, "Analyzing the effects of road type and rainy weather on fuel consumption and emissions: a mesoscopic model based on big traffic data," *IEEE Access*, vol. 9, pp. 62298–62315, 2021.

[10] N. D. Rivera-Campoverde, J. L. Muñoz-Sanz, and B. d. V. Arenas-Ramírez, "Estimation of pollutant emissions in real driving conditions based on data from OBD and machine learning," Sensors, vol. 21, no. 19, Art. 6344, 2021.

[11] E. Giannakis, D. Serghides, S. Dimitriou, and G. Zittis, "Land transport CO_2 emissions and climate change: evidence from Cyprus," International Journal of Sustainable Energy, vol. 39, no. 7, pp. 634–647, 2020.

[12] J. Gan, L. Li, Q. Xiang, and B. Ran, "A prediction method of GHG emissions for urban road transportation planning and its applications," Sustainability, vol. 12, no. 24, Art. 10251, 2020.

[13] V. V. Georgatzi, Y. Stamboulis, and A.Vetsikas, "Examining the determinants of CO_2 emissions caused by the transport sector: Empirical evidence from 12 European countries, "Economic Analysis and Policy, vol. 65, pp. 11–20, Nov. 2020.

[14] Y. Ismail, A. D. Nurwidyawati, and A. D. Rahayu, "Estimation of vehicles carbon dioxide (CO_2) emission, Journal of Industrial Engineering: Scientific Journal on Research and Application of Industrial System,vol. 4, no. 2, pp. 94–99, Sept. 2019.

[15] V. V. Georgatzi, Y. Stamoulis, and A. Vet sikas, "Examining the determinants of CO_2 emissions caused by the transport sector: Empirical evidence from 12

European countries," Economic Analysis and Policy, vol. 65, pp. 11–20, 2020.