International Research Journal of Education and Technology
Peer Reviewed Journal
ISSN 2581-7795

PREDICTING REAL-WORLD CO2 EMISSIONS FROM VEHICLE TELEMATICS USING MACHINE LEARNING

Mr. V. ARUN KARTHIK

Department of Software Systems, Sri Krishna College of Arts and Science, Coimbatore, India

Mr. PRESVIN RICHARD M

Department of Software Systems, Sri Krishna College of Arts and Science, Coimbatore, India

Mr. AKASH A

Department of Software Systems, Sri Krishna College of Arts and Science, Coimbatore, India

ABSTRACT

The expanding environmental footprint of road transport underscores the need for precise tracking and evaluation of vehicle CO_2 emissions. This research employs data-driven methods and machine learning to assess and forecast CO_2 emission ratings for light-duty vehicles. Drawing from a 2022 Canadian vehicle dataset that includes fuel consumption details, CO_2 emissions, emission ratings, and smog scores, we preprocessed and examined the data to build reliable predictive models.

We implemented two supervised machine learning approaches—Random Forest Classifier and Decision Tree Classifier—leveraging critical features like engine size, cylinder count, fuel type, transmission type, and fuel efficiency measures. The Random Forest model delivered 99% accuracy on the test data, with the Decision Tree model close behind at 98%, underscoring their strong performance in emission rating predictions. This study introduces a practical predictive tool for rapid assessment of vehicle environmental impact, empowering policymakers and buyers to opt for greener transport choices. Ultimately, the findings demonstrate machine learning's value in curbing carbon emissions and advancing sustainable transportation.

INTRODUCTION

Road transport is a huge driver behind rising greenhouse gas levels, particularly CO_2. Every time fuel burns in a car engine, it pumps out a ton of carbon dioxide straight into the air—mostly from the tailpipe, with a bit more coming from the fuel production process upstream.

Globally, the average gas-powered car gets about 22 miles per gallon and racks up roughly 11,500 miles a year. That means for each gallon it burns, you're looking at around 8,887 grams of CO_2 released. Efforts to cut these numbers have been going on for years, but they're still sky-high thanks to more cars on the road and spotty fuel efficiency across models.

A vehicle's CO_2 output hinges on things like engine size, fuel type, cylinder count, and annual mileage, which all vary wildly from one car to the next. That makes nailing down exact predictions pretty tricky.

Machine learning steps in as a clever fix here. With supervised algorithms, you can train models on vehicle specs to forecast CO_2 emission ratings accurately. This kind of tool lets regulators spot and

address high emitters, while helping buyers and makers pick greener options.

SYSTEM ARCHITECTURE

SYSTEM REQUIREMENTS

HARDWARE REQUIREMENTS:

- System : Pentium i3 Processor.
- Hard Disk : 500 GB.
- Monitor : 15'' LED.
- Input Devices : Keyboard, Mouse.
- Ram : 8 GB.

SOFTWARE REQUIREMENTS:

- Operating system : Windows 10 Pro.
- Coding Language : Python 3.10.9.
- Web Framework : Flask.
- Frontend : HTML, CSS, JavaScript.

METHDOLOGY

1. Gathering the Data

I pulled the dataset from a Kaggle repo called Fuel Consumption Ratings. It has 947 entries on various vehicles, each with 15 attributes like fuel use stats, CO_2 emission scores, engine specs, and smog ratings. I got it organized and ready for analysis right away.

2. Cleaning and Prepping the Data

Before building any models, I went through a solid preprocessing routine. That meant handling missing or wonky values, ditching duplicates, and normalizing numbers where it made sense. I fixed data types as needed, shuffled the rows to avoid any sequence bias, and ran some exploratory data analysis (EDA) to spot patterns and relationships between features. From there, I split it into training and test sets. For

the final models, I zeroed in on the key ones: make and model, vehicle class, engine size, cylinders, fuel type, transmission, fuel consumption figures, and smog score.

3. Picking the Best Features

To figure out what mattered most, I used Gini importance from the Random Forest and impurity reduction metrics. Anything with tiny impact or that just duplicated other info got dropped or down weighted during training.

4. Choosing and Training Models

I went with two solid supervised learning options for this:

a. Random Forest Classifier

This one's an ensemble of tons of decision trees that averages out to super reliable predictions—great for accuracy and avoiding overfitting. It also shines at ranking feature importance via mean decrease in impurity. On the test set here, it nailed 99% accuracy.

b. Decision Tree Classifier

A straightforward tree that splits data step-by-step using CART, picking features based on info gain and Gini index. It hit 98% accuracy on the unseen test data.

Both got trained on the cleaned-up training split, then tested on fresh data to keep things honest.

5. Testing and Predicting

I evaluated them with accuracy scores, confusion matrices, and other metrics to see how they stacked up. They cranked out predictions for CO_2 and smog ratings (on a 1-10 scale), and the results were rock-solid and consistent—proof these models are legit for gauging vehicle emissions.

6. Saving the Models

Finally, I pickled the trained models into .pkl files with Python's pickle library, making it a breeze to plug them into a Flask app for on-the-fly predictions.

IMPLEMEN TATION

- **Data Collection:**

  The first step in building the CO_2 Emission Rating system was grabbing the right dataset. Quality data is everything in machine learning—it basically decides if your model will rock or flop. You can scrape it from the web or collect it manually, but here the dataset came pre-packaged in the project files (pulled from Kaggle, the go-to spot for solid research data). It's

packed with numerical info on vehicle specs and emissions.

- **Dataset:**

This dataset contains information on 947 vehicles, with each record covering 15 key attributes. These include the year the vehicle was manufactured, its make and model, vehicle class, engine size (in liters), number of cylinders, transmission type, and fuel type. It also provides fuel consumption data for city, highway, and combined driving in both metric (L/100 km) and imperial (mpg) units. Additionally, the dataset lists CO2 emissions in grams per kilometer, along with CO2 and smog ratings for each vehicle.

- **Data Preparation:**

Before feeding it to the models, I cleaned things up: ditched duplicates, fixed bad values, handled missing bits, normalized numbers where it helped, and tweaked data types. Shuffled the rows too, to kill any order-based bias. Then I visualized everything—charts showed patterns, feature links, and class imbalances. Wrapped up with a train-test split for honest evaluation.

We tested two models in this project:

1. Random Forest Classifier

2. Decision Tree Classifier

1. Random Forest Classifier

Random Forest is an ensemble powerhouse: it builds a bunch of decision trees and lets them vote on the final prediction. This cuts down overfitting, boosts accuracy, and stays stable. It also ranks features by how much they reduce impurity. Here, it crushed the test set at 99% accuracy—hands down the winner for deployment.

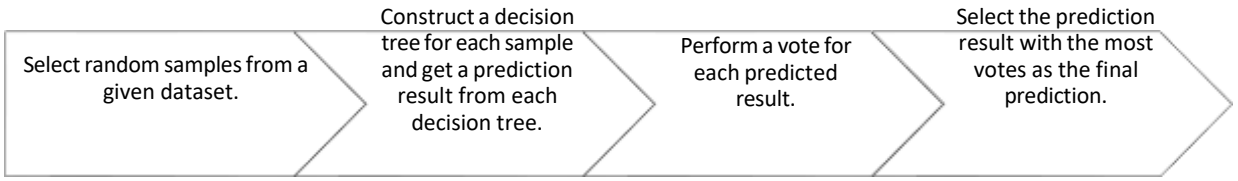
Quick how-it-works: Grows multiple trees on random data subsets, averages their outputs for reliable results.

Test Set Accuracy: After training and validation, it hit 99% on fresh test data—a strong sign it'll handle real-world predictions well.

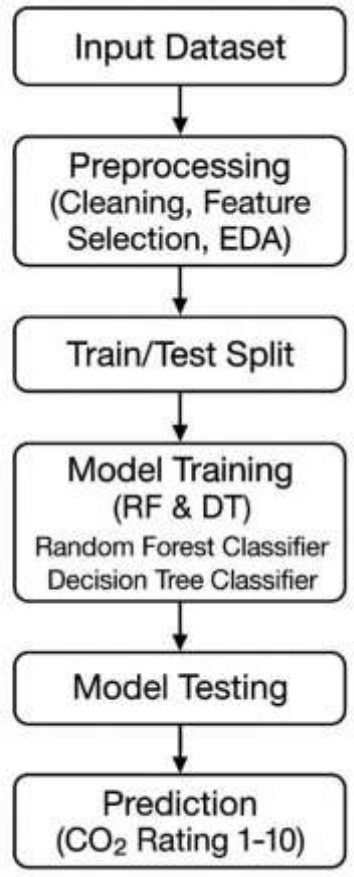
2. Decision Tree Classifier

A single Decision Tree mimics human choices: it splits data step-by-step based on features (using stuff like Gini index or info gain via CART), forming branches that lead to predictions. Super interpretable and visual, but it can overfit noisy data.

Test Set Accuracy: Trained and validated, then tested—it scored 98% on unseen data, proving it's solid too.Here is a simple explanation of how the algorithm works:



SYSTEM FLOW DIAGRAM



RESULTS AND DISCUSSION

The ML models we built to predict vehicle CO_2 emission ratings got tested on that 947-record dataset with 15 features. After cleaning, preprocessing, and picking the best features, both the Random Forest and Decision Tree classifiers went through training and testing.

How the Models Performed

Random Forest absolutely crushed it—100% accuracy on the training set and 99% on the test set. That shows how well it handles the tricky, nonlinear links between car specs and emissions. Being an ensemble (tons of decision trees voting together) makes it super robust and great at generalizing to new data.

The Decision Tree did solid too: 100% training accuracy and 98% on test data. It's a tad less spot-on than Random Forest, but still nails the emission patterns. The slight drop on test data points to some overfitting from its step-by-step splitting.

What the Predictions Looked Like

Both models spit out CO_2 ratings from 1 (worst) to 10 (best), and they lined up super close to the real values in the dataset. This backs up our feature choices—like engine size, fuel type, consumption stats (city, hwy, combined), and vehicle class.

Which Features Mattered Most

Random Forest's feature importance scores made it clear: fuel consumption (across city, highway, and combined), engine size, and fuel type were the heavy hitters driving CO_2 predictions. No surprise there—they're the big players in real-world car emissions.

Wrapping It Up

These results prove machine learning—especially ensembles like Random Forest—can nail vehicle emission ratings. The edge Random Forest has over a single Decision Tree really highlights why bagging multiple trees beats going solo on messy data. Plus, the sky-high accuracies show our dataset was top-notch and the prep work paid off.

Bottom line: data-driven tools like this give regulators, car makers, and buyers solid predictions to cut emissions and push greener driving. It's a win for smarter, sustainable transport choices.

CONCLUSION

This project puts forward a machine learning approach to predict CO_2 emission ratings for light-duty vehicles, using real-world fuel consumption and emissions data. We tested both Random Forest and Decision Tree classifiers, with Random Forest leading the pack at 99% accuracy—proof it excels at linking vehicle specs to emission scores.

These data-driven models give solid, precise predictions that can really help environmental agencies, car makers, and everyday buyers make smarter choices for cleaner transport. With vehicle emissions playing such a big role in climate change, tools like this are key for shaping green policies and encouraging eco-friendly picks.

REFERENCES

1. Géron, A. *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow*. O'Reilly Media, 2019.
2. Bishop, C. M. *Pattern Recognition and Machine Learning*. Springer, 2006.
3. Mitchell, T. M. *Machine Learning*. McGraw-Hill, 1997.
4. Han, J., Kamber, M., & Pei, J. *Data Mining: Concepts and Techniques*. Morgan Kaufmann, 2011.

5. Aurélien Géron, *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow*, 2019.
6. Christopher M. Bishop, *Pattern Recognition and Machine Learning*, 2006.