Predicting Fuel Efficiency Using Tensor Flow in Python

BY:-

Anish Parhi - 2241016181

Soumya Ranjan Behera - 2241002115

Baivab Mishra - 2241013140

Spandan Behera - 2241002116

Jeeban Jyoti Rout - 2241018154

Department of Computer Science & Engineering Faculty of Engineering & Technology (ITER)Jagamohan Nagar, Jagamara, Bhubaneswar, Odisha - 751030

CONTENTS

1:OBJECTIVE:-

Aim of the project:

Our main objective was to predict the vehicle’s fuel efficiency or the MPG(Miles per gallon).

Scope of the Project:

The scope of the project is to develop a Machine Learning Model that gives better attributes i.e., properties of a vehicle for fuel efficiency.

2:INTRODUCTION:-

- The automotive industry which is extremely competitive.
- With increase fuel prices and picky consumers, automobile makers are constantly optimizing their processes to increase fuel efficiency.
- But, what if you could have a reliable estimator for a car’s mpg given some known specifications about the vehicle?
- Then, you could beat a competitor to marker by both having a more desirable vehicle that is more efficient, reducing wasted costs and gaining large chunks of the market.
- Utilizing machine learning, we build prediction model designed to give an edge over company competitor.

3:LITERATURE SURVEY:-

Strengths:

Flexibility and Scalability: TensorFlow provides a flexible and scalable platform for building and deploying deep learning models, making it well-suited for fuel efficiency research that may involve large datasets and complex modeling tasks.

Transfer Learning: TensorFlow supports transfer learning techniques, which can be beneficial for fuel efficiency research by leveraging pre-trained models on related tasks, such as vehicle dynamics or driver behavior analysis, potentially improving model performance and reducing training time.

Limitations:

Data Availability and Quality: The performance of TensorFlow-based models heavily relies on the availability and quality of training data. Fuel efficiency research may be limited by the lack of comprehensive and well-labeled datasets, particularly for specific vehicle types, driving conditions, or geographical regions.

Overfitting and Generalization: Deep learning models can be prone to overfitting, where the model performs well on the training data but fails to generalize to unseen scenarios. Careful regularization techniques and validation strategies are necessary to ensure the robustness and generalization capabilities of fuel efficiency models.

Potential Application:

Vehicle Design Optimization: TensorFlow-based models can be used to optimize vehicle designs for improved fuel efficiency by predicting the impact of different design parameters, such as aerodynamics, weight distribution, and powertrain configurations.

Predictive Maintenance and Diagnostics: Deep learning models can be trained on vehicle sensor data and maintenance records to predict and diagnose issues that may impact fuel efficiency, enabling proactive maintenance and repair strategies to optimize vehicle performance.

4:DESIGN AND METHODOLOGY:-

1. MODULE 1:

Data Collection and training using Machine Learning Algorithms.

2. MODULE 2:

    Preprocessing data

MODULE 1:- Data Collection

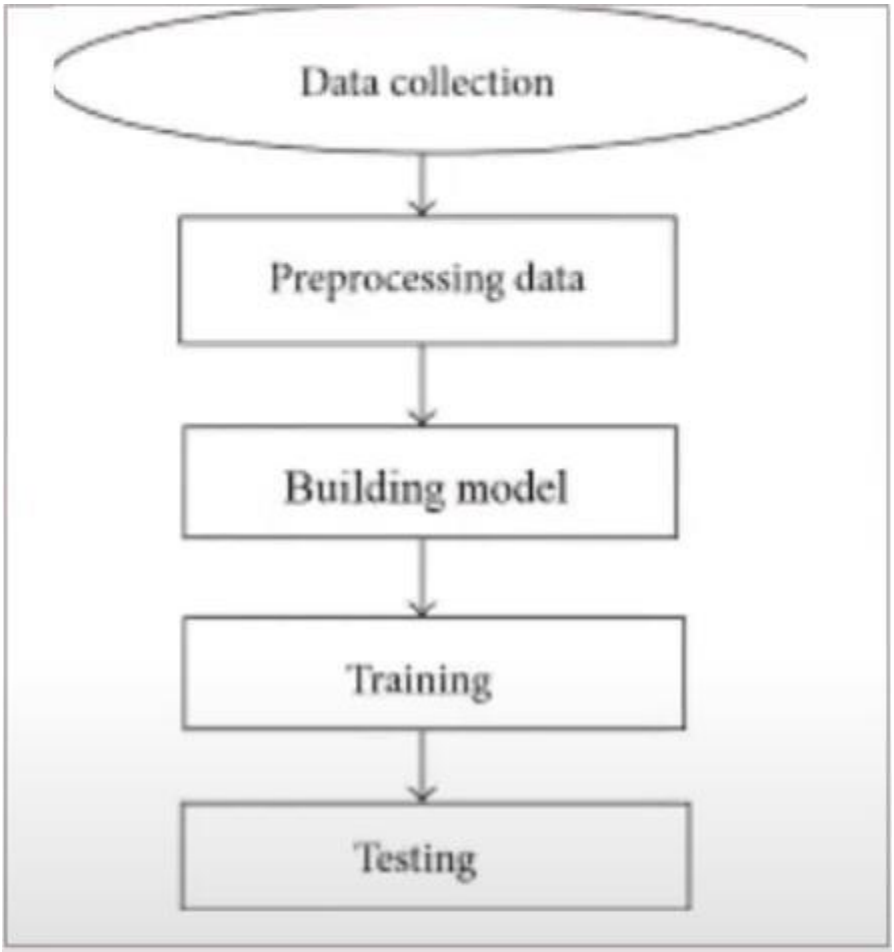
The data set used for this project currently information about the vehicle from which a set of samples have been collected as the training set to get accurate outputs.

We get the data with following attributes:
Mpg: continous
Cylinders: multi-valued discrete
Displacement: continous
Horsepower: continous
Weight: continous
Acceleration:continous
Model year:multi-valued discrete
Origin:multi-valued discrete

MODULE 2:- Preprocessing of data

Here is our dataset there are some null values. So we have done the data processing to make the dataset free from null values.
We plot some box plots to find the median and properties like minimum, first quartile, median, third quartile and maximum.

Data Flow Diagram:-



TESTING:-

- **Unit testing:**
- Unit testing focuses on testing each module indivisually and it also ensures the functionalityof each unit.
- In this testing, all the modules interfaces are verified for the consistency with design specification.
- All error handling paths are also tested.

4.1:METHODOLOGY:-

Matplotlib: It is a versatile plotting library in Python that offers a wide range of visualization options, including static, interactive, and 3D plots. It allows for customization and creation of various types of plots like line plots, histograms, scatter plots.

Numpy: It is a Python library for efficient numerical operations on large arrays and matrices. It provides support for a wide range of mathematical functions and is widely used in scientific computing and data analysis.

Pandas: It is a powerful Python library for data manipulation and analysis. It provides efficient data structures and data analysis tools, making it a popular choice for working with tabular, multidimensional, potentially heterogeneous, and labeled data. Pandas excels at importing, cleaning, transforming, and analyzing complex datasets.

Seaborn: It is a powerful Python data visualization library built on top of Matplotlib. It provides a high-level interface for creating attractive and informative statistical graphics, including scatter plots, line plots, bar plots, heatmaps, and more. Seaborn excels at visualizing complex datasets and highlighting statistical relationships.

TensorFlow: It is a free and open-source machine learning library known for building deep neural networks with high-level Python code. Developed by Google, it offers APIs for training, analyzing, and deploying ML models across various platforms.

4.1.2:Data Collection and Preprocessing:

- Collect relevant data related to fuel efficiency, such as vehicle specifications (e.g., engine size, weight, aerodynamics), driving conditions (e.g., speed, acceleration, road conditions), and actual fuel consumption measurements.
- Preprocess the data by handling missing values, outliers, and categorical variables.

4.1.3:Feature Engineering:

- Analyze the collected data and identify relevant features that may influence fuel efficiency.
- Apply feature scaling or normalization techniques, if necessary.

4.1.4:Model Architecture Selection:

- Evaluate various neural network architectures, such as Feedforward Neural Networks (FNNs), Convolutional Neural Networks (CNNs), or Recurrent Neural Networks (RNNs), based on the nature of the input data and the problem complexity.
- Utilize TensorFlow's built-in layers and model-building utilities to construct the chosen neural network architecture.

4.1.5:Model Training:

- Define the loss function (e.g., mean squared error) and the optimization algorithm (e.g., Adam optimizer) for training the model.
- Leverage TensorFlow's automatic differentiation capabilities for efficient gradient computation and model optimization.

4.1.6:Model Evaluation and Tuning:

- Evaluate the trained model's performance on the testing dataset using appropriate metrics (e.g., mean absolute error, root mean squared error).
- Analyze the model's predictions and compare them with the actual fuel efficiency values.

4.1.7:Continuous Improvement:

- Monitor the performance of the deployed fuel efficiency model in real-world scenarios and collect feedback for further improvements.
- Periodically retrain the model with new data or update the model architecture as new techniques or insights become available.

5.IMPLEMENTATION:-

•**Data Collection and Preprocessing**: We load the data from a CSV file, handle missing values, and one-hot encode categorical features using pandas.

•**Feature Engineering**: We define the numerical and categorical input features, and create input tensors for each feature using Keras functional API.

•**Model Architecture Selection**: We define a Feedforward Neural Network architecture using Keras functional API, with Dense layers, Dropout, and Concatenate layers.

•**Model Training**: We compile the model with the mean squared error loss and Adam optimizer, and train the model using the fit() method with early stopping callback.

•**Model Evaluation and Tuning**: We evaluate the model's performance on the test set using mean absolute error (MAE) metric. We also use the ELI5 library to interpret the model's predictions and analyze the importance of different input features using permutation importance.

•**Model Deployment and Integration**: We save the trained model to an HDF5 file for deployment.

•**Continuous Improvement**: We include a comment indicating the need for monitoring and retraining the model with new data for continuous improvement.

```
[ ] dataset = raw_dataset.copy()
    dataset.tail()
```

	MPG	Cylinders	Displacement	Horsepower	Weight	Acceleration	Model Year	Origin
393	27.0	4	140.0	86.0	2790.0	15.6	82	1
394	44.0	4	97.0	52.0	2130.0	24.6	82	2
395	32.0	4	135.0	84.0	2295.0	11.6	82	1
396	28.0	4	120.0	79.0	2625.0	18.6	82	1
397	31.0	4	119.0	82.0	2720.0	19.4	82	1

```python
dataset = pd.get_dummies(dataset, columns=['Origin'], prefix='', prefix_sep='')
dataset.tail()
```

	MPG	Cylinders	Displacement	Horsepower	Weight	Acceleration	Model Year	Europe	Japan	USA
393	27.0	4	140.0	86.0	2790.0	15.6	82	False	False	True
394	44.0	4	97.0	52.0	2130.0	24.6	82	True	False	False
395	32.0	4	135.0	84.0	2295.0	11.6	82	False	False	True
396	28.0	4	120.0	79.0	2625.0	18.6	82	False	False	True
397	31.0	4	119.0	82.0	2720.0	19.4	82	False	False	True

```python
train_dataset.describe().transpose()
```

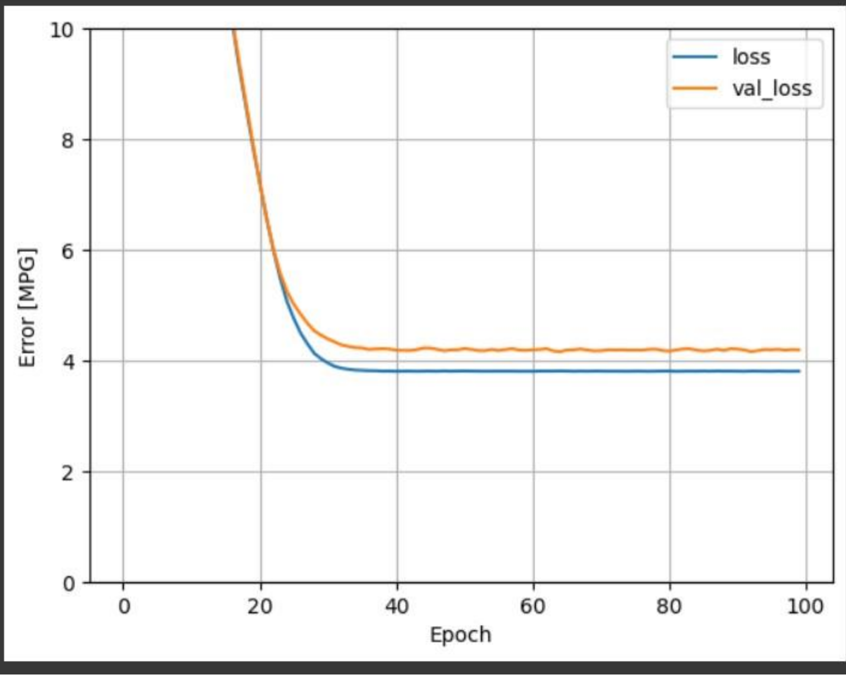
	count	mean	std	min	25%	50%	75%	max
MPG	314.0	23.310510	7.728652	10.0	17.00	22.0	28.95	46.6
Cylinders	314.0	5.477707	1.699788	3.0	4.00	4.0	8.00	8.0
Displacement	314.0	195.318471	104.331589	68.0	105.50	151.0	265.75	455.0
Horsepower	314.0	104.869427	38.096214	46.0	76.25	94.5	128.00	225.0
Weight	314.0	2990.251592	843.898596	1649.0	2256.50	2822.5	3608.00	5140.0
Acceleration	314.0	15.559236	2.789230	8.0	13.80	15.5	17.20	24.8
Model Year	314.0	75.898089	3.675642	70.0	73.00	76.0	79.00	82.0

```python
train_dataset.describe().transpose()[['mean', 'std']]
```

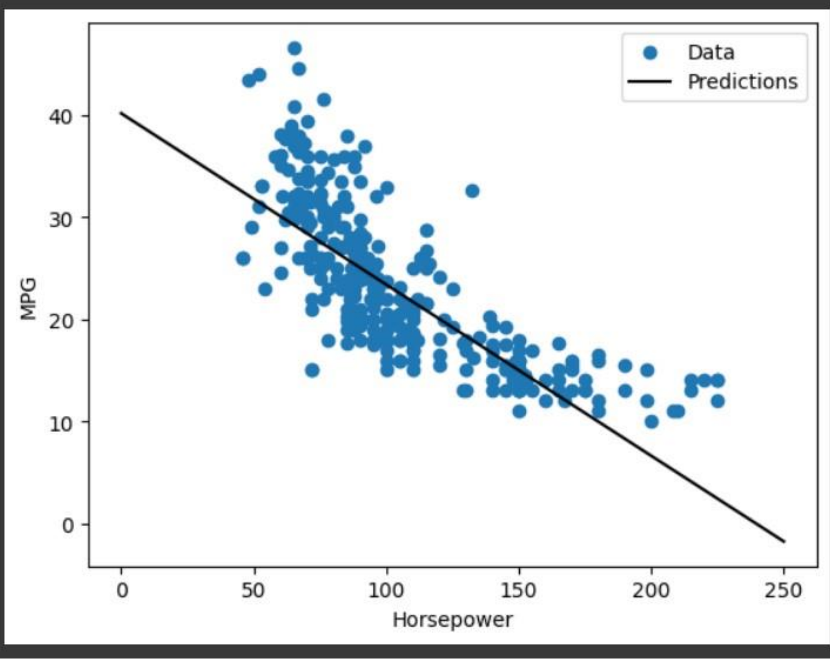
	mean	std
MPG	23.310510	7.728652
Cylinders	5.477707	1.699788
Displacement	195.318471	104.331589
Horsepower	104.869427	38.096214
Weight	2990.251592	843.898596
Acceleration	15.559236	2.789230
Model Year	75.898089	3.675642

6.OUTPUT:-

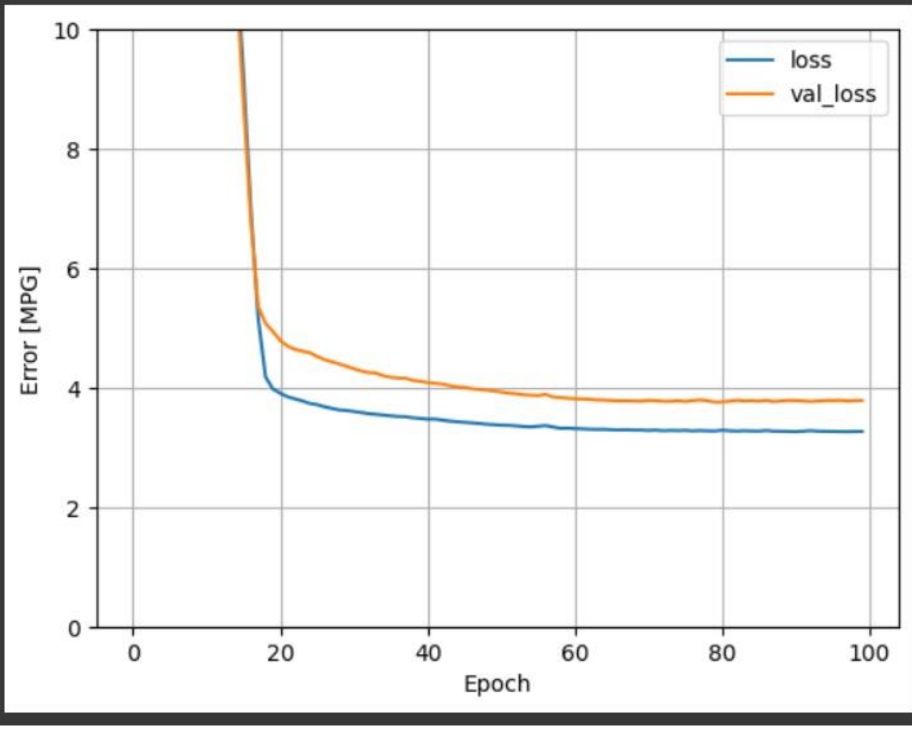
```
plot_loss(history)
```



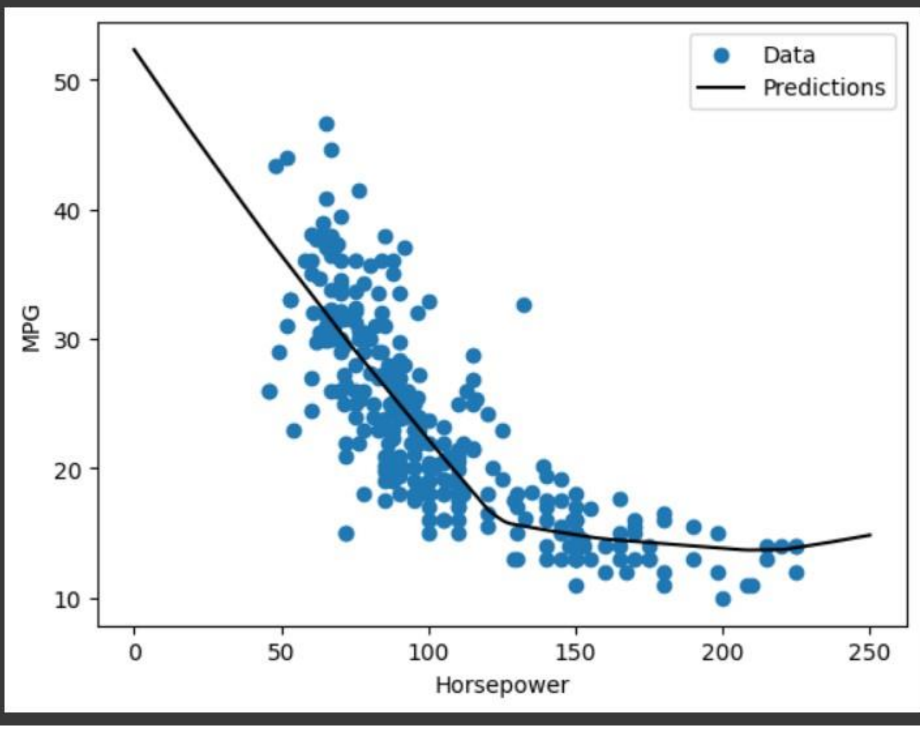

```
plot_horsepower(x, y)
```



```
plot_loss(history)
```




```
plot_horsepower(x, y)
```



7.CONCLUSION:-

Fuel prices are increasing rapidly each day, and the demand of vehicles with better fuel efficiency or Miles per gallon is growing tremendously.

During this project, our main objective was to predict the vehicle's fuel efficiency or the MPG(Miles per gallon). We have done the data preprocessing to make the dataset free from null values and other disturbances, then we performed data visualization of the data represent and know well about the attributes in the dataset.
We have implemented various machine learning models and checked for their errors and accuracies until we get the best effective model for the data taken.

Fuel-efficient cars emit less pollution over the same amount of distance traveled. They also cost less to operate. Buying a more fuel-efficient car can save you thousands of dollars on fuel costs over time and can often balance out a higher purchase prices if you keep the car long enough.

8.REFERENCES:-

1. Allcot, H. (October 2009). "Social Norms and Energy Conservation."

2. Prepared for the Center for Energy and Environmental Policy Research, Massachusetts Institute of Technology. Barkenbus, J. N. (2010).

3. "Eco-driving: An Overlooked Climate Change Initiative." Energy Policy (38:2); pp. 762–769. Boriboonsomsin, K.; Vu, A.; Barth, M. (2010).

4. Eco-Driving: Pilot Evaluation of Driving Behavior Changes. Riverside, CA: College of Engineering—Center for Environmental Research and Technology, University of California Riverside.

5. Earleywine, M.; Gonder, J.; Markel, T.; Thornton, M. (September 2010). "Simulated Fuel Economy and Performance of Advanced Hybrid Electric and Plug-in Hybrid Electric Vehicles Using In-Use Travel Profiles." Proceedings of the 6th IEEE Vehicle Power and Propulsion Conference (VPPC). Lille, France.

6. Ehrhardt-Martinez, K. (April 2010). "Policy Innovations and People: Active Participants in the Energy Revolution." Prepared for the ACEEE 30th Anniversary Symposium: Energy and the Economic Imperative, April 2010. Washington, D.C.

7. SAE International. (September 2010). "Utility Factor Definitions for Plug-In Hybrid Electric Vehicles Using Travel Survey Data." Surface Vehicle Information Report SAE J2841.