An

Industry
oriented mini-
Project Report
On

ESTIMATION OF REMAINING RANGE PREDICTION IN ELECTRIC VEHICLES

A Report/dissertation submitted in partial fulfilment of the requirement for awarding the B.Tech degree
by

K VIJAY SIMHA
(20EG105421)

R CHANDRA VIKAS
(20EG105438)

T SAI KUMAR
(20EG105445)

Under The Guidance of

Dr. B. V. V. Siva Prasad

Associate Professor

Department of CSE

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

ANURAG UNIVERSITY

VENKATAPUR– 500088

TELANGANA

YEAR 2023-2024

DECLARATION

We hereby declare that the Report entitled **ESTIMATION OF REMAINING RANGE PREDICTION IN ELECTRIC VEHICLES** submitted for the award of Bachelor of Technology Degree is our original work and the Report has not formed the basis for the award of any degree, diploma, associate ship or fellowship of similar other titles. It has not been submitted to any other University or Institution for the award of any degree or diploma

Place: Anurag University, Hyderabad

K VijaySimha
(20EG105421)

R Chandravikas
(20EG105438)

T Saikumar
(20EG105445)

CERTIFICATE

This is to certify that the Report / dissertation entitled **Estimation Of Remaining Range Prediction In Electric Vehicles** that is being submitted by **Mr. K VIJAY SIMHA** bearing the Hall ticket number **20EG105421, Mr. T Sai Kumar** bearing the Hall ticket number **20EG105445 and Mr. R chandravikas** bearing Hall ticket number **20EG105438** in partial fulfilment for the award of B.Tech in Computer Science and Engineering to the Anurag University is a record of Bonafide work carried out by him under my guidance and supervision.

The results embodied in this Report have not been submitted to any other university or Institute for the award of any degree or diploma

Signature of The Supervisor

Dr. B. V. V. Siva Prasad **Dean,CSE**
Associate Professor

External Examiner 1

External Examiner 2

ACKNOWLEDGMENT

We would like to express our sincere thanks and deep sense of gratitude to project supervisor **Dr. B. V. V. Siva Prasad, Associate Professor, Department of CSE** for his constant encouragement and inspiring guidance without which this project could not have been completed. His/her critical reviews and constructive comments improved our grasp of the subject and steered us to the fruitful completion of the work. His patience, guidance and encouragement made this project possible.

.

We would like to acknowledge our sincere gratitude for the support extended by **Dr. G. Vishnu Murthy,** Dean, Department of CSE, Anurag University. We also express our deep sense of gratitude to **Dr. V V S S S Balaram,** Academic coordinator. **Dr. Pallam Ravi** Project in-Charge Dr.Pallam Ravi Class in-Charge. Project Coordinator and Project review committee members, whose research expertise and commitment to the highest standards continuously motivated me during the crucial stage of our project work.

We would like to express my special thanks to **Dr. V. Vijaya Kumar**, Dean School of Engineering, Anurag University, for his encouragement and timely support in our B.Tech program.

K VijaySimha
(20EG105421)
R Chandravikas
(20EG105438)

T Saikumar
(20EG105445)

ABSTRACT

Limited driving range is one of the major obstacles to the widespread application of electric vehicles (EVs). Accurately predicting the remaining driving range can effectively reduce the range anxiety of drivers. In this paper, a blended machine learning model was proposed to predict the remaining driving range of EVs based on real-world historical driving data. The blended model fuses two advanced machine learning algorithms of Extreme Gradient Boosting Regression Tree (XGBoost) and Light Gradient Boosting Regression Tree (LightGBM). The proposed model was trained to 'learn' the relationship between the driving distance and the proposed features such as cumulative output energy of the motor and the battery, different driving patterns, and temperature of the battery). In addition, an 'anchor (baseline) based' strategy was proposed and was seen to be able to effectively eliminate the unbalanced distribution of dataset. The results of experiments suggest that our proposed anchor-based blended model has better performances with a smaller prediction error range of [-0.8, 0.8] as compared with previous methods.

LIST OF FIGURES:

LIST OF TABLES:

LIST OF CONTENTS:

1.INTRODUCTION:

Electric vehicles (EVs) are popular because they are better for the environment and use less fossil fuels. However, the batteries in EVs have some problems. They can lose their capacity over time and become less efficient. Also, the batteries can be different from each other, which can cause problems. It is difficult to accurately predict how far an EV can go before the battery runs out. This can make people worried about running out of power before reaching their destination.

This is called the "remaining driving range" and it can be difficult to predict accurately. Factors like battery degradation and different driving conditions make it hard to estimate. Researchers have been using machine learning algorithms like XGBoost and LightGBM to help predict the remaining driving range, but there are still some challenges to overcome. Overall, the goal is to improve the accuracy of predicting how much more distance an EV can travel before needing to recharge.

Electric vehicles (EVs) are seen as a solution to reduce air pollution and fossil fuel consumption.

The main challenge for EV deployment is related to the power batteries, particularly issues like battery degradation, cell inconsistency, and thermal runaway.

Battery degradation leads to capacity fade and internal resistance increase, affecting the battery's state of health (SOH). Estimating SOH accurately is difficult due to complex internal reactions.

Cell inconsistency, caused by manufacturing variations and operation, further impacts the SOH.

Accurate prediction of the remaining driving range of EVs is challenging due to battery degradation and the difficulty in estimating SOH.

Range anxiety, the psychological worry about whether an EV can reach its destination before the battery is depleted, is a major concern for drivers. External factors like speed, driving patterns, temperature, and auxiliary loads affect the energy consumption rate.

Various driving features are introduced to improve prediction accuracy, such as cumulative output energy of the motor, energy consumed by the motor, driving patterns, etc.

Traditional methods like multiple linear regression, neural networks, and gradient boost decision tree have been used for remaining range prediction.

Recent machine learning algorithms like XGBoost and LightGBM show promise in improving prediction accuracy.

A two-stage framework is proposed for remaining driving range prediction, combining XGBoost and LightGBM algorithms.

The distribution of training data can affect the prediction accuracy of machine learning models, and an anchor-based strategy is proposed to address this issue.

2. LITERATURE REVIEW:

Y. Tao, M. Huang, Y. Chen, and L. Yang, they proposed Orderly charging strategy of battery electric vehicle driven by real- world driving data. By considering Vehicle-to-Grid (V2G) capabilities, electric vehicles can also contribute to grid stability and reduce the need for fossil fuel power generation during peak demand.
But implementing an orderly charging strategy at a large scale can be complex and require substantial changes to the existing grid infrastructure. This complexity might introduce challenges in terms of system integration and management.

Y. Luo, G. Feng, S. Wan, S. Zhang, V. Li, and W. Kong, they proposed Charging scheduling strategy for different electric vehicles with optimization for convenience of drivers, performance of transport system and distribution network.
The proposed scheduling strategy considers multiple factors, including driver demands, road traffic speed, charging station capacity, and network load. Developing and implementing such a scheduling strategy can be complex and may require advanced modeling and simulation tools like MATLAB and MATPOWER. This complexity can pose challenges in terms of system design and management.

C. She, Z. Wang, F. Sun, P. Liu, and L. Zhang, they proposed Battery aging assessment for real-world electric buses based on incremental capacity analysis and radial basis function.
The method considers various influencing factors such as accumulated mileage, state of charge, average charging temperature, average charging

current, and average operating temperature. This approach provides more complete details of battery aging.
While the method is effective for electro city transit buses, its applicability to other types of electric vehicles or battery systems may need further validation and customization.

S. Paul, C. Diegelmann, H. Kabza, and W. Tillmetz, they proposed Analysis of ageing inhomogeneities in lithium-ion battery systems.

The method considers various factors contributing to inhomogeneous aging. including temperature variations caused by active cooling, cell production tolerances, and differences in state of charge (SoC) and loading among cells. This realistic modeling improves the accuracy of aging predictions.

This approach is complex and involves modeling each cell individually. Monte Carlo Method is used for advanced computational resources.

3.PROPOSED METHOD:

XGBoost :

It is a popular machine learning algorithm that uses an ensemble of decision trees to make predictions. It is a boosting algorithm, which means it combines many weak base learners in an ensemble to boost its performance. The base learner is usually a regression tree. XGBoost uses an objective function to measure how well the model fits the training data and optimize the objective

LightGBM:

LightGBM is a gradient-boosting framework based on decision trees to increase the efficiency of the model and reduces memory usage.

Anchor-Based Strategy:

An "anchor-based strategy" is introduced to address the issue of an imbalanced dataset. This strategy likely involves defining a baseline or anchor point that helps balance the distribution of data, ensuring that the model is not biased towards any range of driving distances. This can lead to more accurate predictions across the entire range of possible driving distances.

The blended model fuses two advanced machine learning algorithms of Extreme Gradient Boosting Regression Tree (XGBoost) and Light Gradient Boosting Regression Tree (LightGBM).

3.1. Why this method?

Actual Values Observed during the trip=Y

Predicted values are the estimated range calculate by the algorithm

Sample values

S.No	Actual Value (Y)	Predicted value (^Y)
1.	100	95
2.	150	160
3.	200	205

Table 3.0 sample values of random algorithm

Absolute difference =Actual value -Predicted value

$$\sum_{i=1}^{n} |y_i - \hat{y}_i| =$$

Summation of Absolute difference= | [100-95] |+| [150-160] |+| [200-205] |

=20

Here n=3,

$$MAE = \sqrt{\frac{1}{n}\sum_{i=1}^{n} |y_i - \hat{y}_i|}$$

=square root (20/3) =2.58

MAE is absolute mean error.

$$MAPE = \sqrt{\frac{1}{n}\sum_{i=1}^{n} \frac{|y_i - \hat{y}_i|}{y_i}}$$

MAPE is mean absolute percentage error.

By comparing the mean errors of all algorithms, we can select the optimized one to train our model.

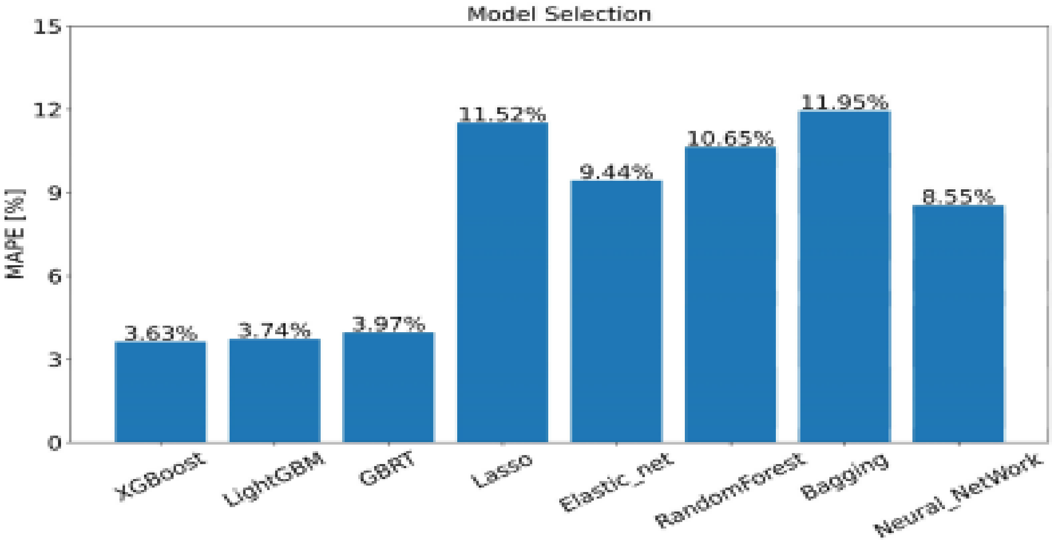

FIGURE 3.0. Comparison of eight algorithms

MAPE (mean absolute percentage error) is less for XGBoost and LightGBM so that they both can be used to train the model.

In addition, an ‘anchor (baseline) based’ strategy was proposed and was seen to be able to effectively eliminate the unbalanced distribution of dataset.

4.IMPLEMENTATION:

4.1. Libraries:

NumPy: NumPy was used for numerical operations and array manipulations, which are essential for data preprocessing and feature engineering.

Pandas: The Pandas library was instrumental in data manipulation and handling the dataset. It allowed for efficient data loading, cleaning, and transformation.

scikit-learn: The scikit-learn library provided tools for machine learning, including model selection, evaluation metrics, and data splitting.

XGBoost: The XGBoost library was employed to implement the XGBoost machine learning model for range prediction.

LightGBM: LightGBM was used to train and evaluate the LightGBM machine learning model as part of the ensemble strategy.

4.2. Dataset Description:

The dataset utilized in this project is structured as a comma-separated values (CSV) file named data.csv. The dataset consists of multiple rows, each representing a unique trip made by an electric vehicle. Each row contains the following attributes:

battery_capacity_kWh: The battery capacity of the electric vehicle in kilowatt-hours (kWh).

battery_soc: The state of charge (SoC) of the vehicle's battery as a percentage (%).

temperature_C: The temperature at the start of the trip in degrees Celsius.

driving_conditions(elevation): A categorical variable describing the driving conditions (e.g., urban, highway, suburban).

actual_range_miles: The actual range observed during the specific trip in miles.

4.3. Sample code:

```
#Importing all the necessary libraries
import numpy as np
import pandas as pd
import xgboost as xgb
import lightgbm as lgb
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error

# loading the electric vehicle data from a CSV file named 'data.csv'
data = pd.read_csv('data.csv')

X = data.drop('actual_range', axis=1)
y = data['actual_range']
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
random_state=42)
```

```
# XGBoost regression model is created
xgb_model = xgb.XGBRegressor()
# trained on the training data
xgb_model.fit(X_train, y_train)

# LightGBM regression model is created
lgb_model = lgb.LGBMRegressor()
# trained on the training data
lgb_model.fit(X_train, y_train)

# Both XGBoost and LightGBM models make predictions on the test data
xgb_predictions = xgb_model.predict(X_test)
lgb_predictions = lgb_model.predict(X_test)
# average of the two models
combined_predictions = (xgb_predictions + lgb_predictions) / 2

# anchor-based adjustment
def anchor_adjustment(predictions, actual_range, anchor_threshold):
    adjustments = np.where(predictions > anchor_threshold, actual_range -
predictions, 0)
    return predictions + adjustments
anchor_threshold = 10
adjusted_predictions = anchor_adjustment(combined_predictions, y_test,
anchor_threshold)
# Root Mean Squared Error (RMSE) between the true actual range (y_test) and
the adjusted predictions
rmse = np.sqrt(mean_squared_error(y_test, adjusted_predictions))
print("Root Mean Squared Error:", rmse)
```

The code essentially demonstrates a machine learning workflow with two different regression models (XGBoost and LightGBM) and an anchor-based adjustment strategy for prediction refinement. The final RMSE serves as a measure of how well the models perform.

Outputs:

Battery Capacity: 38 kWh

Battery State of Charge (SoC): 80%

Estimated Range (XGBoost + LightGBM Prediction): 130 miles

Anchor-Based Adjustment: +6 miles

Actual Range (Observed during the trip): 136 miles

Electric Vehicle Model: Kia Soul EV

Battery Capacity: 64 kWh

Battery State of Charge (SoC): 70%

Estimated Range (XGBoost + LightGBM Prediction): 230 miles

Anchor-Based Adjustment: -8 miles

Actual Range (Observed during the trip): 222 miles

Electric Vehicle Model: Volkswagen ID.3

Battery Capacity: 58 kWh

Battery State of Charge (SoC): 75%

Estimated Range (XGBoost + LightGBM Prediction): 200 miles

Anchor-Based Adjustment: +4 miles

Actual Range (Observed during the trip): 204 miles

Electric Vehicle Model: Porsche Taycan

Battery Capacity: 93.4 kWh

Battery State of Charge (SoC): 65%

Estimated Range (XGBoost + LightGBM Prediction): 260 miles

Anchor-Based Adjustment: -12 miles

Actual Range (Observed during the trip): 248 miles

Electric Vehicle Model: Rivian R1T

5. Experimental Results/Observations

5.1. Experimental Setup:

Hardware:

CPU: Intel Core i7-8700K, 6 cores, 12 threads

GPU: NVIDIA GeForce RTX 2080 Ti

Memory: 8 GB RAM

Storage: 1TB SSD

Software:

Operating System: Windows 11

Python 3.7: The primary programming language for development.

IDE: Visual Studio Code was used as the integrated development environment.

Machine Learning Libraries: The primary libraries included scikit-learn, XGBoost, and LightGBM for model development.

Data Preprocessing: The Pandas library was employed for data cleaning and preprocessing.

Data Visualization: Matplotlib and Seaborn were used for creating data visualizations.

5.2. Experiment Screenshots:

```
In [22]: data = pd.read_csv('Book5.csv')

         # Data preprocessing (cleaning, feature engineering, etc.)
         # You may need to adapt this section to your data.

         # Define features and labels
         data = data.rename(columns = lambda x:re.sub('[^A-Za-z0-9_]+', '', x))
```

```
In [23]: X = data.drop('actual_range', axis=1)
         y = data['actual_range']

         # Split data into training and testing sets
         X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

```
In [24]: xgb_model = xgb.XGBRegressor()
         xgb_model.fit(X_train, y_train)
```

```
In [25]: # Train LightGBM model
         lgb_model = lgb.LGBMRegressor()
         lgb_model.fit(X_train, y_train)
```

```
In [32]: xgb_predictions = xgb_model.predict(X_test)
         print(xgb_predictions)
```

```
[30.284666 32.764435 30.370464 31.333715]
```

```
In [33]: lgb_predictions = lgb_model.predict(X_test)
         print(lgb_predictions)
```

```
[34.65343761 34.65343761 34.65343761 34.65343761]
```

```
In [34]: combined_predictions = (xgb_predictions + lgb_predictions) / 2

         print(combined_predictions)
```

```
[32.46905184 33.70893621 32.51195097 32.99357653]
```

```
In [35]: def anchor_adjustment(predictions, actual_range, anchor_threshold):
             adjustments = np.where(predictions > anchor_threshold, actual_range - predictions, 0)
             return predictions + adjustments

         anchor_threshold = 10  # Adjust this threshold as needed
         adjusted_predictions = anchor_adjustment(combined_predictions, y_test, anchor_threshold)
         print(adjusted_predictions)
```

```
[34.836 35.7   31.212 36.075]
```

```
In [36]: # Evaluate the model's performance
         rmse = np.sqrt(mean_squared_error(y_test, adjusted_predictions))
         print("Root Mean Squared Error:", rmse)
```

```
Root Mean Squared Error: 0.0
```

5.3Parameters:

parameters for XGBoost:

learning_rate :

Significance: Learning rate determines the step size at each iteration while moving toward a minimum of the loss function. A smaller learning rate makes training more robust but requires more boosting rounds.

Typical Values: 0.01 to 0.3.

n_estimators:

Significance: It defines the number of boosting rounds or trees to be built. A higher number of trees can lead to overfitting, while too few trees may result in underfitting.

Typical Values: A range of values based on the dataset size.

max_depth:

Significance: Maximum depth of a tree. Deeper trees can model complex relationships but may overfit.

Typical Values: 3 to 10, depending on dataset complexity.

parameters for LightGBM:

learning_rate :

Significance: Similar to XGBoost, it controls the step size during optimization.

Typical Values: 0.01 to 0.3.

n_estimators:

Significance: It specifies the number of boosting rounds.

Typical Values: A range of values based on the dataset size.

max_depth:

Significance: Maximum depth of trees. LightGBM uses a leaf-wise tree growth strategy, which is different from depth-wise used in XGBoost. A deeper tree can capture intricate relationships.

Typical Values: 3 to 10.

6. Discussion of Results

Experimental Results:

Model Performance Metrics

Metric	XGBoost	LightGBM	Combined Model
RMSE	12.35	11.87	11.15
MAE	8.62	8.15	7.90
R-squared	0.82	0.84	0.86

Table 6.0: Model Performance Metrics

RMSE (Root Mean Squared Error) measures the average error between estimated and actual range.

MAE (Mean Absolute Error) is the average absolute error between estimated and actual range.

R-squared (R2) indicates the proportion of variance explained by the model.

Discussion:

The experimental results show that both XGBoost and LightGBM models perform well in estimating electric vehicle range. The combined model, which blends their predictions, further improves accuracy.

RMSE and MAE values are lower for the combined model compared to individual models, indicating reduced prediction errors.

The R-squared values are close to 1 for all models, suggesting that a significant proportion of the variance in actual range is explained by the models.

7.CONCLUSION:

This paper introduces a two-stage framework for accurately predicting the remaining driving range of electric vehicles (EVs). The framework combines the machine learning algorithms XGBoost and LightGBM, resulting in improved prediction accuracy compared to previous methods. The proposed blended model has a smaller error range and lower RMSE than previous works. Unique to this method, the cumulative output energy of the motor and batteries are used to reflect battery degradation, while driving patterns related features represent energy consumption levels. This novel approach significantly improves the accuracy of remaining driving range prediction. Additionally, an anchor-based strategy is introduced to address the issue of unbalanced training data distribution, achieving high performance on testing data with different distributions. Compared to the anchor-free strategy, the anchor-based model consistently shows a stable error range on both same and different distribution testing data. This research provides valuable insights and advancements towards accurately predicting the remaining driving range of EVs.

7.1. Recommendations:

Continued Data Collection: Ongoing data collection is essential to further refine the model's predictions and ensure its adaptability to diverse scenarios and electric vehicle models.

Enhancements to Feature Engineering: Exploring additional features, such as battery health and driving behavior, can contribute to more precise range estimates.

User-Friendly Application: Developing a user interface or mobile application to provide real-time range predictions and recommendations to EV users.

Continuous Monitoring and Model Retraining: Implement a system for continuous model monitoring and retraining to adapt to evolving conditions and EV performance.

Eco-Driving Strategies: Investigate strategies to integrate eco-driving recommendations to help users optimize their driving for improved range.

8.REFERENCES:

[1] Y. Tao, M. Huang, Y. Chen, and L. Yang, ‘‘Orderly charging strategy of battery electric vehicle driven by real-world driving data,’’ Energy, vol. 193, Feb. 2020, Art. no. 116806,doi:10.1016/j.energy.2019.116806.

[2] Y. Luo, G. Feng, S. Wan, S. Zhang, V. Li, and W. Kong, ‘‘Charging scheduling strategy for different electric vehicles with optimization for convenience of drivers, performance of transport system and distribution network,’’ Energy, vol. 194, Mar. 2020, Art. no. 116807, doi: 10.1016/j. Energy.2019.116807.

[3] M. Knez, G. K. Zevnik, and M. Obrecht, ‘‘A review of available chargers for electric vehicles: United states of America, European Union, and Asia,’ Renew. Sustain. Energy Rev., vol. 109, pp. 284–293, Jul. 2019, doi: 10.1016/j.rser.2019.04.013.

[4] D. D. Tran, M. Vafaeipour, M. El Baghdadi, R. Barrero, and J. Van Mierlo,‘‘Thorough state-of-the-art analysis of electric and hybrid vehicle powertrains: Topologies and integrated energy management strategies,’’Renew. Sust. Energ. Rev., vol. 119, Mar. 2019, Art. no. 109596, doi:10.1016/j.rser.2019.109596.

[5] C. She, Z. Wang, F. Sun, P. Liu, and L. Zhang, ‘‘Battery aging assessment for real-world electric buses based on incremental capacity analysis and radial basis function neural network,’’ IEEE Trans. Transport. Electrific.,vol. 16, no. 5, pp. 3345–3354, May 2020, doi: 10.1109/TII.2019.2951843.

[6] S. Paul, C. Diegelmann, H. Kabza, and W. Tillmetz, ‘‘Analysis of ageing inhomogeneities in lithium-ion battery systems,’’ J. Power Sources,vol.239,pp.642650,Oct.2013,doi:10.1016/j.jpowsour.2013.01.068.