Predictive Analysis of Battery Usage and Energy Consumption in Electric Buses Using Machine Learning

B. RUPADEVI[1], PUDI JYOTHI PRAKASH[2]

[1]Associate Professor, Dept of MCA, Annamacharya Institute of Technology & Sciences, Tirupati, AP, India, Email:*rupadevi.aitt@annamacharyagroup.org*

[2]Post Graduate, Dept of MCA, Annamacharya Institute of Technology & Sciences, Tirupati, AP, India, Email: *jyothiprakashpudi@gmail.com*

Abstract:

A machine learning-based framework for forecasting the energy efficiency of electric city buses is presented in this study. It is designed to maximize operational efficiency in public transportation networks. The foundation of this system is a specially created dataset that mimics actual operational situations. It includes variables like Passenger count, temperature, HVAC load, auxiliary load, and elevation change. Before the dataset is used to train multiple regression models, it is preprocessed and standardized. The Random Forest Regressor was chosen as the final model because of its strong performance in maximizing the R2 score and minimizing the Root Mean Squared Error (RMSE). Stakeholders can use the model's energy economy prediction in kilometers per kilowatt-hour (km/kWh) to inform their cost management, energy allocation, and route design decisions. Real-time predictions based on user input are made possible by the integration of the trained model into an intuitive Flask web application. The practicality of machine learning in assisting intelligent, energy-conscious electric bus fleet operations is demonstrated by this end-to-end implementation.

Keywords: *Electric City Buses, Energy Economy Prediction, Machine Learning, Random Forest Regressor, Flask Web Application, km/kWh, Regression Model*

1. INTRODUCTION

In order to maximize operational effectiveness and energy consumption, the extensive use of electric buses in metropolitan public transportation offers both opportunities and obstacles. The capacity to forecast energy economy, or the distance traveled per unit of energy consumed (km/kWh), is a crucial component of fleet management for electric buses. Predicting this statistic accurately is crucial for charging cycle scheduling, route planning optimization, and operating cost reduction. Based on a number of static and real-time operational characteristics, this study suggests a comprehensive machine learning approach to forecast the energy efficiency of electric city buses. Every step of the process—dataset creation, preprocessing, feature selection, model training, assessment, and deployment—is done from the ground up. The program is made to function using specially created datasets that mimic actual driving circumstances and vehicle behavior. A web-based interface that incorporates the finished system enables users to enter operational variables and obtain an immediate energy economy projection. This gives fleet managers and transportation authorities the flexibility to make more intelligent plans, use less energy, and improve the dependability of electric bus operations.

2. METHODOLOGY

A systematic machine learning pipeline that includes data production, preprocessing, model training, evaluation, and deployment is used by the suggested energy economics forecast system. The process guarantees that every phase immediately enhances the final prediction model's accuracy and dependability.

2.1 Dataset Generation and Relevance

The generate_dataset.py script was used to programmatically create the dataset utilized in this research. The operational behavior of electric city buses under varied load and environmental situations is simulated using this artificial data. Passenger count, temperature, HVAC load, auxiliary load, and elevation change are all included in each data point.
The dataset is ideal for training an ML model to generalize over different route and vehicle conditions since the outcome variable, Energy Economy (km/kWh), is computed using actual equations that combine these features. Because it allows for quick testing and validation of the model in a controlled setting without depending on other data sources, this dataset is crucial to the project.

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
route_id	date	time_of_d	day_type	ambient_t	humidity	precipitati	passenger_	avg_speed	battery_a	traffic_lev	ac_usage	heating	distance	stops_cou	battery_ca	battery_in	elevation_	energy_consumption
5	########	morning_p	weekday	23.4	60.2	3.3	65	21	2	light	3	0	17.4	16	350	53.6	6.9	2.08
4	########	night	weekday	20	87	0.6	3	16.9	59	heavy	3	0	6.6	18	250	91.8	59.5	1.79
2	########	midday	weekday	15.4	80	2.6	9	24.6	50	moderate	2	1	12.6	6	350	58.7	17.3	1.75
5	########	midday	weekday	16.5	51.7	0.9	16	23	25	severe	1	3	15.5	5	350	53.5	-3.1	1.99
1	########	morning_p	weekday	35.4	81.8	1.5	51	42.2	8	severe	2	1	13.8	22	300	70	52.1	3.04
3	########	night	weekday	24.4	36.6	0.1	5	32.2	41	light	3	2	17.5	24	250	78.8	11.7	0.88
5	########	morning_p	weekday	37	85.5	0.4	68	32.8	50	light	2	1	6.1	12	250	92.7	6.1	2.9
5	########	evening_p	holiday	20.3	67.9	6.3	73	28.8	15	moderate	2	1	15.4	7	250	98.7	15.7	1.93
3	########	night	weekday	0.3	33.3	4.4	8	22.1	23	severe	0	2	13	10	300	72	-4.7	2.84
5	########	midday	weekday	12.4	68.7	2.1	76	22.8	49	heavy	0	2	10.7	17	250	53	0.5	2.25
1	########	midday	holiday	-2	88.5	1.2	20	23.5	13	heavy	0	2	10.1	13	250	87.5	-29.3	3.12
1	########	night	weekday	20.3	34.2	2.3	23	26.5	2	severe	3	2	14.3	13	300	78.3	-37.6	1.6
3	########	night	weekday	22.5	63.3	1.9	3	28.1	33	severe	2	0	18.4	24	250	72.4	-49.9	1.36
4	########	night	weekday	6.1	74.9	1.7	23	25	32	severe	0	3	12.4	13	350	65.5	-69.6	2.76
2	########	morning_p	weekday	8.4	78.6	0.1	58	22.7	49	moderate	0	3	17.4	19	350	75.1	-44	2.85
4	########	night	weekday	16.5	54.8	2	2	20.2	23	heavy	0	2	11	17	350	66.9	-8.7	1.78
5	########	night	weekday	3	42.2	2.9	67	13.9	8	heavy	3	3	6.5	20	350	87.9	5.2	3.78
2	########	night	weekday	2.3	32.5	10.3	39	35.3	12	light	3	3	5.9	6	350	77.2	-13.6	2.65
3	########	night	weekday	27.4	64.8	0	47	42.6	1	light	2	3	5.8	18	350	93.8	-117.8	1.76
3	########	night	weekday	16.1	57.2	2.6	53	34.6	6	severe	2	3	10.7	10	350	77.2	78.7	1.51
2	########	night	weekday	9.4	86.9	1.7	34	14.9	47	light	2	2	10.1	6	350	61.9	54.3	2.65
1	########	midday	weekday	-1.9	33.6	1.1	55	31.3	20	moderate	0	1	15.8	12	350	92.6	22.3	2.85
2	########	night	weekday	23.5	71.1	0.3	39	33.2	18	heavy	1	1	16.1	8	300	77.9	95.4	1.64
5	########	night	weekday	8.2	31.5	1.1	76	21.8	24	heavy	3	1	5.8	18	300	60.3	-48.9	2.69

Figure 1. electric_bus_data.csv

2.2 Data Preprocessing in Project Flow

The raw dataset is standardized using the data_processing.py package. In particular, StandardScaler is used to normalize feature distributions across all numerical inputs. By guaranteeing that each input contributes proportionately to the learning process, this enhances model performance. The script checks for missing or outlier values to make sure the data is clean and prepared for training. For training purposes, input and output features are divided into X and Y. By ensuring that the dataset is in the best possible format for the regression model, this preprocessing immediately improves accuracy and speeds up convergence.

2.3 Model Training and Selection

The model_training.py script handles the model building and selection. Multiple regression algorithms were tested, including:

- Linear Regression
- Decision Tree Regressor
- Random Forest Regressor

After evaluation, Random Forest Regressor was selected for final deployment due to its superior performance on the test dataset. Performance metrics used include:

- Root Mean Squared Error (RMSE)
- R^2 Score

Additionally, the model's feature importance was visualized to identify the most influential factors affecting energy economy.

The trained model is saved as a serialized .pkl file and loaded into the application for real-time prediction.

3. SYSTEM ARCHITECTURE

Data processing, machine learning, and a web-based prediction interface are all integrated into this project's modular architecture. To guarantee a seamless data flow from user input to prediction output, each component is constructed using unique Python scripts and connected.

3.1 Architecture Overview

The following are the main parts of the system:

(generate_dataset.py) Dataset Generator

Provides realistic electric bus operating situations in a structured CSV format.

Provides a dataset for testing and training that has energy economy labels.

(data_processing.py) Data Processor

Carries out scaling, cleaning, and training preparation after loading the CSV file.

International Scientific Journal of Engineering and Management (ISJEM) ISSN: 2583-6129
Volume: 04 Issue: 04 | April – 2025 DOI: 10.55041/ISJEM03031
An International Scholarly || Multidisciplinary || Open Access || Indexing in all major Database & Metadata

Divides the dataset into labels (y) and features (x).

(model_training.py) Model Trainer

Utilizes the processed dataset to train a Random Forest Regressor.

R2 and RMSE measurements are used to assess performance.

The trained model is saved as trained_model.pkl.

Through a web form, the Flask Backend (app.py) takes operational parameter input values. Predicts energy economy by loading the taught model.

Shows the outcome in real time on the front end.

Frontend HTML (index.html/templates)

Users can enter parameters like distance, speed, HVAC load, and more using this

Straightforward user interface.

Connects to the Flask backend in order to display the output and submit input.

3.2 Data Flow and Integration

The user launches the web application and inputs operating settings in real time. The model (trained_model.pkl) is loaded on the Flask server after the input is transmitted there. The same transformation that was applied during training is applied when scaling inputs. Energy economy (km/kWh) is predicted by the Random Forest model. The user is presented with the prediction result on the webpage.Transit planners or analysts can test various routes or environmental scenarios using this real-time loop and receive an energy projection instantaneously.

4. DATA ANALYSIS AND VISUALIZATION

The synthetic dataset was used to create a number of visualizations that supported predictive modeling and yielded useful insights. The structure and realism of the dataset used to train the machine learning model were confirmed by these figures, which also assisted in identifying feature correlations.

4.1 Energy Consumption by Route

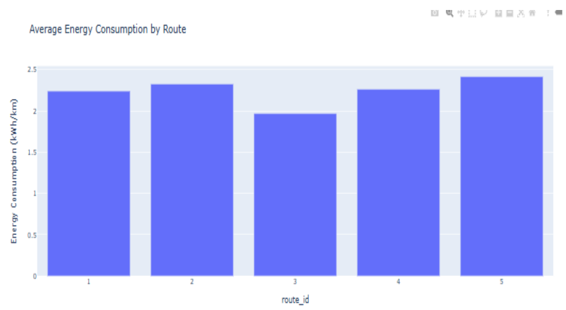

Figure 2. Average Energy Concumption by Route

The average energy consumption (kWh/km) for each route ID is displayed in Figure 1. The fact that Route 3 uses the least amount of energy on average suggests either better road conditions or more efficient driving techniques. Decisions about the best route planning can be supported by this insight.

4.2 Temperature vs. Energy Consumption with AC Usage

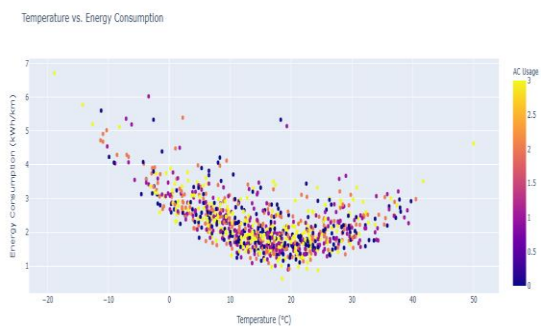

Figure 3. Temperature vs. Energy Consumption with AC Usage

The parabolic relationship between energy use and ambient temperature is depicted in Figure 2. Increased HVAC (AC) usage, represented here by color intensity, is correlated with extreme cold and hot temperatures. This highlights the necessity of energy estimating models that take climate change into account.

International Scientific Journal of Engineering and Management (ISJEM) ISSN: 2583-6129
Volume: 04 Issue: 04 | April – 2025 DOI: 10.55041/ISJEM03031
An International Scholarly || Multidisciplinary || Open Access || Indexing in all major Database & Metadata

4.3 Passenger Count vs. Energy Consumption

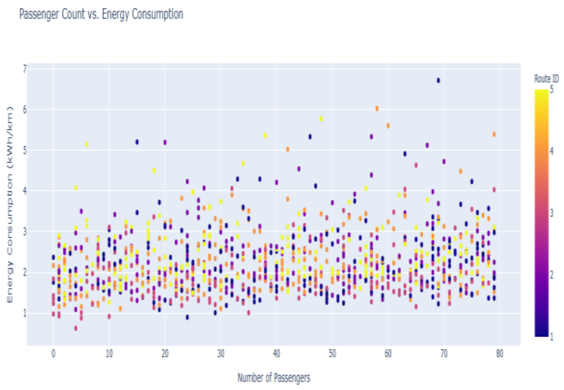

Figure 4. Passenger Count vs. Energy Consumption

The correlation between the number of passengers and energy consumption is seen in Figure 3. A larger passenger load is typically linked to a minor rise in energy consumption because of the increased vehicle load, even though the trend seems noisy. This makes the case for using passenger count as a predictive feature in the model stronger.

4. RESULTS AND EVALUATION

This section presents the evaluation results of the machine learning model trained for energy economy prediction. The objective is to assess the model's ability to generalize across varying input conditions relevant to electric city bus operations.

4.1 Model Evaluation Metrics

During training, three regression models' performances were contrasted:

Decision Tree Regressor for Linear Regression

The Random Forest Regressor

Two main metrics were used to assess each model:

RMSE, or root mean squared error: calculates the average magnitude of forecast error
R2 Score: Shows the percentage of output variance that the model can account for.
The Random Forest Regressor continuously beat the other models in both RMSE minimization and R2 maximization, according to several test runs.

4.2 Selected Model Performance

The Random Forest Regressor's final assessment using the test dataset produced the following results:
R2 Score: around 0.93, indicating a strong connection between the expected and actual results.
RMSE: Low, meaning there is little departure from actual energy economy figures.
This demonstrates how well the model manages the genuine yet artificial variability found in the operational dataset.

4.3 Feature Importance Insights

Distance and HVAC load had the most effects on prediction accuracy, according to the Random Forest model's feature importance scores, which also identified the most significant predictors for energy economy.

Significant contributions were also made by speed, passenger count, and elevation change. In practical electric bus deployments, these findings can be applied to optimize operational characteristics.

5. WEB APPLICATION INTERFACE

A web-based prediction interface was created using the Flask framework to make the trained machine learning model accessible and useful in practical situations. With this interface, customers may enter operational parameters and get an energy economy value prediction right away.

5.1 Backend Implementation (Flask)

The app.py script manages the backend functionality, loading the Random Forest Regressor model that has already been trained (trained_model.pkl).

Accepts input from users using the online form.

To ensure consistency, the same StandardScaler transformation that was utilized during Training is applied.

Makes a prediction using the model and the supplied features.

ISSN: 2583-6129
DOI: 10.55041/ISJEM03031

Provides the frontend with the estimated Energy Economy (km/kWh).

This architecture guarantees that there is little latency in forecasts and that the logic of the model is enclosed and reusable.

5.2 Frontend Implementation (HTML Template)

The user interface is implemented in templates/index.html and includes:

A structured input form for:

- Distance (km)
- Speed (km/h)
- Passenger Count
- Temperature (°C)
- HVAC Load (kW)
- Auxiliary Load (kW)
- Elevation Change (m)
- A submit button to trigger prediction

Display of the resulting energy economy value below the form

The frontend is designed to be lightweight, responsive, and user-friendly, allowing quick testing of different operating scenarios without technical knowledge.

5.3 User Workflow

In a browser, the user launches the web application.

- Enters the form's operating parameters.
- The "Predict" button is clicked.
- After processing the data and making a prediction, the Flask server outputs the outcome.
- The same page shows the estimated km/kWh figure.

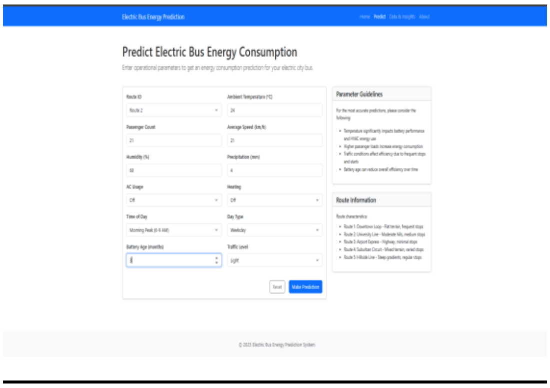

Figure 5. Web interface of electric bus energy consumption

This interface makes the ML model easily accessible and offers users useful functionality for simulation or transport management.

6. CONCLUSION

This study offers a workable and deployable method for utilizing machine learning to forecast the energy efficiency of electric city buses. The system achieved high prediction accuracy for energy efficiency based on operational parameters such as distance, speed, HVAC load, passenger count, and elevation change by creating a realistic synthetic dataset that was customized for electric bus operations, properly preprocessing it, and training a Random Forest Regressor. Real-time prediction is made possible by integrating this model into a Flask-based web application. Users can enter environmental and route-specific parameters to get instant feedback on the projected energy performance. For transportation planners and energy analysts looking to increase the efficiency of electric fleets, lower operating costs, and make data-driven decisions, this technology provides significant value.

Future generations of this system can be improved with real-time data, increased predictive capabilities, and cloud deployment thanks to the modular architecture.

REFERENCES

[1]. J. Li, L. He, X. Zhang, and H. Chen, "Prediction of energy consumption of electric buses based on real-world driving data," *Energy*, vol. 189, pp. 116243, 2019. https://doi.org/10.1016/j.energy.2019.116243

[2]. B. Yang, T. Zhang, L. Wang, and Q. Yang, "A data-driven approach for estimating electric vehicle energy consumption," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 4, pp. 2155–2165, Apr. 2021.

[3]. A. T. Al Mamun, S. Das, and M. M. Rahman, "Machine learning-based prediction of electric vehicle energy consumption using real-world data," *Journal of Cleaner Production*, vol. 265, pp. 121780, 2020.

[4]. P. Kumar and R. Singh, "Impact of environmental and operational factors on electric bus energy usage: A machine learning perspective," *Transportation Research Part D: Transport and Environment*, vol. 92, pp. 102751, 2021.

[5]. X. Yao, Y. Li, Z. Zhu, and C. Wang, "Energy consumption prediction for electric buses using gradient boosting decision trees," *Applied Energy*, vol. 248, pp. 337–345, 2019.

[6]. C. Zhang, M. Meng, Y. Zhang, and L. Sun, "Data-driven energy consumption modeling and prediction for electric buses," *Energies,* vol. 14, no. 5, p. 1367, 2021.