ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue X, October 2026

Predicting Electric Bus Energy Consumption Using Explainable Machine Learning

Rahul Madeti, C. K. Revanth Kumar Reddy, Rajat, Dodda Cholika, Khasim Syed*

School of Computer Science and Engineering (SCOPE), VIT-AP University, Amaravati, Andhra Pradesh 522237, India

***Corresponding Author**

DOI: https://doi.org/10.51583/IJLTEMAS.2026.151000078

Received: 24 October 2026; Accepted: 29 October 2026; Published: 07 November 2026

ABSTRACT

Battery electric buses (BEBs) must finish every trip on a fixed store of energy, so a poor trip-level forecast costs either a stranded bus or a needless charging stop. This paper presents a reproducible pipeline that predicts trip energy intensity (kWh/km) from seven pre-trip variables, explains each prediction with SHAP (SHapley Additive exPlanations), and passes the forecast to a pre-dispatch range-assurance check and a time-of-use depot-charging scheduler, all served through a 13-endpoint REST interface. Real fleet telemetry was not available, so the study uses two physics-based synthetic benchmarks: a 10,459-trip RouteZero-style set (RZ-Syn) and an 8,500-trip Indian-conditions set (IN-Syn). Nineteen physically motivated features feed four regressors. On six held-out buses (1,864 trips), histogram gradient boosting reaches MAE 0.0374 kWh/km, RMSE 0.0541 kWh/km, R^2 0.9916 and MAPE 3.45%, against a generator noise ceiling of R^2 0.9949; over 20 re-drawn bus splits MAPE is 3.44 ± 0.08%, and the Indian benchmark gives R^2 0.9829. Stress tests temper these figures: withholding the hillside regime raises MAPE to 6.43%, transfer between benchmarks raises it above 20% with systematic under-prediction, and a per-variable input-range check does not reliably detect either. The nominal 95% interval covers only 86.1% of test trips; split-conformal recalibration restores 94.5%. In a low-battery dispatch test the worst-case range check missed none of 735 infeasible trips, whereas a static average-consumption rule missed 10.3%. Smart charging cut simulated electricity cost by 32–71%. Validation on measured fleet data is the necessary next step.

Index Terms: battery electric bus, energy consumption prediction, gradient boosting, SHAP, explainable machine learning, range assurance, smart charging, synthetic data

INTRODUCTION

A battery electric bus (BEB) leaves the depot with a fixed amount of energy and no easy way to get more. Packs hold a few hundred kilowatt-hours, a full recharge takes hours, and a bus that runs flat on route means a cancelled service and a recovery operation. How much energy a trip will need is therefore the number the whole day's plan is built around, and it is a hard number to pin down. On a measured Finnish fleet the average was 0.78 kWh/km, yet individual trips differed by more than 1 kWh/km [8]. The causes of that spread (road gradient, passenger load, stop density, and the heating or cooling that temperature forces) interact rather than simply add [2], [3], [19].

The two ways of getting the forecast wrong both cost money. Under-prediction strands buses. Over-prediction makes planners add charging stops and spare vehicles that were never needed. Many operators still plan with a fleet-wide average consumption, which ignores exactly the route-specific factors above, while the more accurate machine-learning models in the literature are often black boxes that a depot manager has little reason to trust. Research on the problem has moved from physics-based vehicle models [2], [19], through machine-learning predictors trained on simulated [3], [4], [7] or measured [5], [8], [9] data, to explainable pipelines that use SHAP (SHapley Additive exPlanations) to show which inputs drive a forecast [5]–[7]. The RouteZero

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue X, October 2026

project [1] pairs a data-driven route-energy model with a depot-charging optimiser, and the pipeline reported here follows the same two-stage logic: forecast first, then plan. What we found harder to locate was a small, self-contained implementation that guards against leakage with an explicit grouped split, explains its predictions, carries the forecast through to dispatch and charging decisions, and says plainly how far its accuracy figures can be trusted. This paper aims to supply one.

One point has to be stated early because it shapes everything after it. We did not have trip-level BEB telemetry. Both datasets used here are produced by physics-based simulators written for this project (Section IV). The accuracy figures therefore measure how well a model recovers a known physical structure from noisy samples; they do not measure performance on real buses. We report a noise ceiling for each dataset, and several stress tests, so that the reader can see what the headline numbers do and do not show. A recent explainable-ML study built on a parametric tropical bus dataset [10] takes the same stance.

The contributions are as follows. (i) A 19-feature, physically motivated representation of a bus trip, with ablations showing which feature groups carry the signal. (ii) A benchmark of four regressors on two datasets under a bus-grouped split, with bootstrap confidence intervals and a noise-ceiling analysis. (iii) SHAP explanations cross-checked against permutation importance. (iv) A range-assurance check and a time-of-use depot scheduler driven by the forecast, evaluated quantitatively and exposed, with the model, through a REST service. (v) An audit of the evaluation itself: repeated grouped splits, leave-regime-out validation, cross-benchmark transfer, an input-range check, interval calibration with a split-conformal remedy, and the artefacts of the data generators. Section II reviews related work, Section III describes the system architecture, Section IV the data, Section V the method, Section VI the results and their discussion, and Section VII concludes.

RELATED WORK

Physics-Based and Measured-Data Models

Basma et al. [2] built a detailed model of the powertrain, HVAC, auxiliaries and battery thermal management of a BEB and showed how driving pattern, weather and passenger load change consumption, with auxiliary and thermal loads sharply reducing range in extreme conditions. Lajunen [19] compared the energy use and cost of hybrid and electric city buses, and Gallet et al. [9] used real-world data to estimate energy demand across large public-transport networks. Vepsäläinen et al. [8] measured six buses on three routes in southern Finland and found that ambient temperature, driving behaviour and route characteristics dominated the variation. In warm climates, heating, ventilation and air conditioning (HVAC) alone can take a sizeable share of the energy budget [10], [20].

Machine-Learning Predictors

Abdelaty et al. [3] compared seven statistical and machine-learning models on a Simulink-generated dataset and found that all explained more than 90% of the variance, with road gradient and initial state of charge (SoC) the most influential inputs; a companion multivariate regression [4] explained more than 96%. Nan et al. [5] chained an LSTM for driving variables to an XGBoost energy model and explained it with SHAP, and a later study [6] combined random forests, SHAP and partial-dependence plots to identify electricity-saving driving behaviours. Wang et al. [7] applied XGBoost and SHAP to a 169,344-scenario simulation dataset, where road gradient, initial SoC and HVAC use emerged as the dominant factors.

Synthetic Data and Evaluation Practice

When telemetry is out of reach, simulated or parametric data are a common substitute [3], [7], [10]. Almajed et al. [10] generated 4,000 physics-informed trip scenarios and reported an XGBoost R^2 of 0.944 and a MAPE of 3.51%, then said plainly that, because the target comes from an equation whose inputs are also the model's features, the result is not live-fleet validation. Their ablations, and a quantile interval whose coverage reached 79.4% against a 90% target, are the kind of self-scrutiny we try to follow. On evaluation design, random splits of grouped data leak information between training and test sets [17], [18], which is why we split by bus. As Section VI-G shows, grouping is necessary but not sufficient when the groups carry no signal.

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue X, October 2026

Integrated Planning

RouteZero [1] couples route energy prediction with depot charging optimisation to limit the load placed on the grid. Standards such as ISO 15118 [21] define the vehicle-to-charger communication that smart-charging schedules of this kind rely on. Our work is a smaller, self-contained implementation of the same idea, with more emphasis on explanation and on honest evaluation.

System Architecture

Fig. 1 shows the prediction pipeline as designed at the project's design review. Raw trip records pass through loading, preprocessing and feature engineering. The feature set, paired with the target (energy per kilometre), is split by group into training, validation and test partitions. Four models of increasing complexity are trained on the same split and compared on MAE, RMSE, R^2 and MAPE, SHAP analysis is applied to the selected model, and the best model is saved with its scaler and feature list behind a single predict_energy() function.

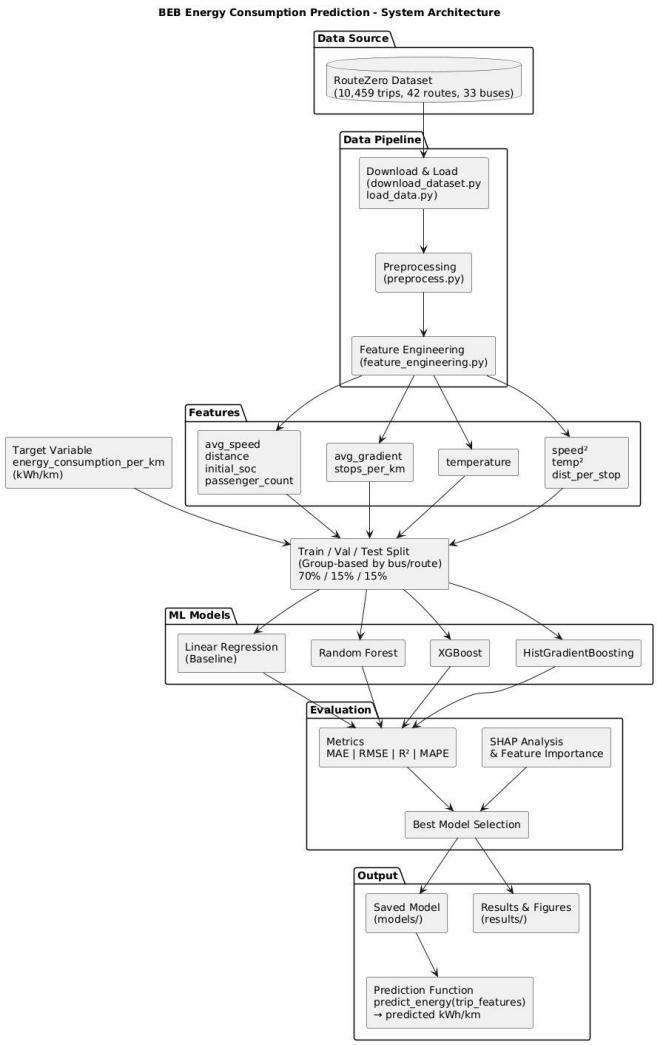


Fig. 1. Pipeline architecture from the project design review. Module names in the diagram correspond to the loader, preprocessor and feature-engineering modules of the final implementation.

Fig. 2 shows what was built on top of that pipeline. A FastAPI service exposes 13 endpoints: health check; single and batch prediction, each with an interval; global and per-trip SHAP attributions and the stored plots; model metrics and metadata; depot optimisation and the tariff table; and single and roster-wide range assurance. Two decision modules consume the forecast. Range assurance decides before dispatch whether a bus can complete a trip with its reserve intact, and the depot scheduler decides when each bus should charge overnight. A small web dashboard, a command-line predictor and the Python function serve as clients.

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue X, October 2026

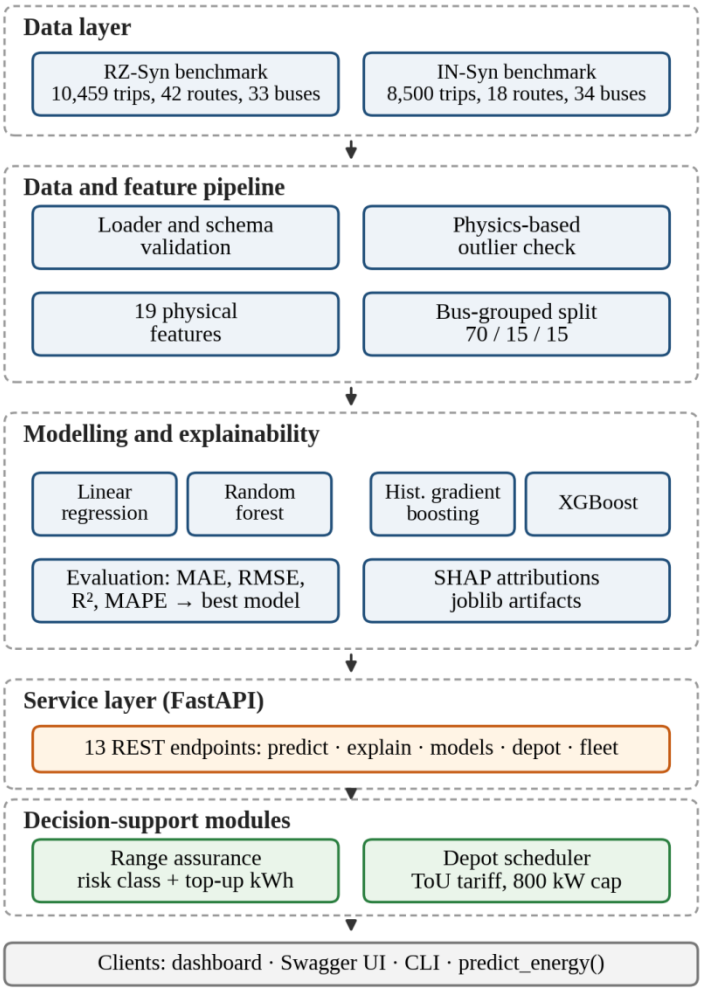


Fig. 2. Deployed system. Both benchmarks feed one pipeline; the selected model and its explanations sit behind a REST service that also hosts the range-assurance and depot-scheduling modules.

DATA AND PREPROCESSING

Provenance

Real trip-level BEB telemetry that combines energy, gradient, load, temperature and state of charge was not part of the project's data. Two physics-based simulators were written instead. The first, RZ-Syn, produces 10,459 trips on 42 routes and 33 buses, the scale given in the project proposal, in the variable layout of Table I. The second, IN-Syn, produces 8,500 trips on 18 routes in seven Indian cities (Bengaluru, Delhi-NCR, Mumbai, Pune, Hyderabad, Shimla and Dehradun–Mussoorie), served by 34 buses of four models. Neither set is RouteZero's measured data and neither should be mistaken for it.

B. Energy Generators

Each trip is assigned a speed, gradient, stop density, load and temperature drawn from route-specific distributions, and its energy rate is then computed from vehicle dynamics. With vehicle mass M, speed v in m/s, gradient G in percent and s stops per kilometre, the net propulsive force and the traction energy per kilometre are

$$F = C_{rr} M g + \tfrac{1}{2}\rho C_d A v^2 + M g \tfrac{G}{100} \qquad (1)$$

$$e_{\text{trac}} = \frac{1000F + s \cdot \tfrac{1}{2} M v^2 (1 - \eta_r)}{3.6 \times 10^6 \eta_d} \qquad (2)$$

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue X, October 2026

where ηr is the regenerative recovery fraction and ηd the drivetrain efficiency; on descents only 45% of the (negative) traction work is credited back. HVAC power follows a V-shaped law around the cabin setpoint Tset, and auxiliary energy per kilometre is the total auxiliary power divided by speed, so slow trips pay more per kilometre:

$$P_{\mathrm{HVAC}} = k\,|T - T_{\mathrm{set}}|^{\beta} \quad (3)$$

$$e = \left(e_{\mathrm{trac}} + \frac{P_{\mathrm{base}} + P_{\mathrm{HVAC}}}{v}\right) f_{\mathrm{SoC}}\,\varepsilon \quad (4)$$

Here fSoC = 1 + 0.003(25 − SoC)1.4 when SoC is below 25% (and 1 otherwise), and ε is multiplicative Gaussian noise. For RZ-Syn: M = 12,500 kg plus 70 kg per passenger, Crr = 0.0095, ρ = 1.205 kg/m³, Cd = 0.65, A = 7.5 m², ηr = 0.52, ηd = 0.88, Pbase = 2.6 kW, Tset = 21 °C, β = 1.32, k = 0.42 below 12 °C and 0.28 otherwise, and ε ~ N(1, 0.035²). Rates are clipped to 0.45–3.85 kWh/km. RZ-Syn trips come from four route archetypes: urban dense (R01–R14), suburban arterial (R15–R28), hillside (R29–R35) and express (R36–R42). Temperature is N(21, 7.5²) clipped to 3–42 °C, and departure SoC mixes morning (mean 92%) and midday (mean 68%) departures.

IN-Syn changes the physics to suit Indian operation: rolling resistance rises with road roughness (Crr = 0.0110 + 0.0008•IRI), passengers weigh 65 kg and can number 85, mass and frontal area depend on whether the bus is 9 m or 12 m, the setpoint is 24 °C, air-conditioning draws up to 14.5 kW and adds a passenger heat load, ηr = 0.49, ηd = 0.865 and the noise has a standard deviation of 3.2%. Road roughness, AC state and bus size are inputs to the simulator but not to the learned models, so IN-Syn contains hidden factors that RZ-Syn does not.

Descriptive Statistics and Generator Artefacts

Table I summarises the model inputs and the target. Fig. 3 shows RZ-Syn: energy climbs almost linearly with gradient, the V-shaped temperature effect is hard to see in the raw scatter because gradient dominates the variance, and the speed relationship is a broad cloud. Three artefacts of the generators matter for what follows. First, 16.1% of RZ-Syn trips (0.4% of IN-Syn) sit exactly at the lower clip, so the target has a spike that real, regenerating buses would not show. Second, SoC falls below 25% in only 7 of 10,459 RZ-Syn trips and in none of IN-Syn, so the battery-resistance mechanism exists in the equations but can hardly be learned. Third, a coding slip in the IN-Syn generator makes departure SoC take just two values, 41.3% and 84.5%. We left the files as generated, because every reported result derives from them, and treat these points as limitations.

TABLE I

INPUT AND TARGET STATISTICS (MEAN ± SD, RANGE)

Variable	RZ-Syn mean ± SD	RZ-Syn range	IN-Syn mean ± SD	IN-Syn range
Distance (km)	15.25 ± 7.13	3.50–44.45	24.37 ± 10.47	4.42–58.00
Gradient (%)	0.45 ± 1.40	−4.49–5.20	0.70 ± 1.57	−2.08–8.50
Passengers	26.6 ± 11.9	0–80	55.1 ± 17.5	5–85
Stops per km	1.66 ± 0.69	0.30–3.89	2.77 ± 0.89	1.10–5.20
Speed (km/h)	24.3 ± 8.0	10.0–55.0	17.8 ± 5.4	8.5–41.0
Initial SoC (%)	79.8 ± 16.0	20.0–100	62.7 ± 21.6†	41.3–84.5†
Temperature (°C)	21.0 ± 7.5	3.0–42.0	26.7 ± 7.9	−2.0–47.5
Energy (kWh/km)	**1.07 ± 0.58**	**0.45–3.85**	**1.61 ± 0.61**	**0.48–4.25**

† Only two distinct values occur (41.3% and 84.5%); see Section IV-C.

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue X, October 2026

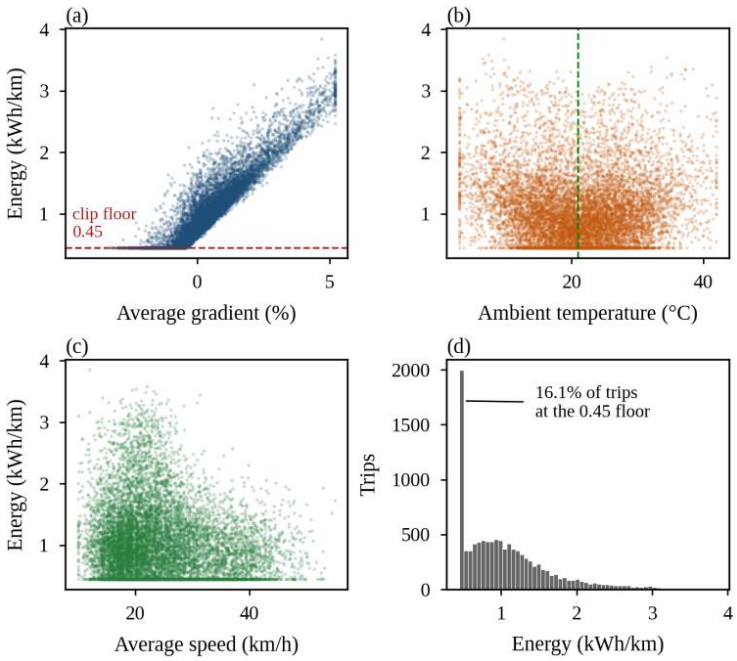


Fig. 3. Exploratory plots of RZ-Syn (all 10,459 trips). (a) Energy versus gradient; the dashed line marks the 0.45 kWh/km clip. (b) Energy versus ambient temperature; the dashed line is the 21 °C setpoint. (c) Energy versus speed. (d) Distribution of the target, with the spike at the clip floor.

Cleaning

The contextual rules remove only physically impossible records: non-positive distance, speed outside 3–85 km/h, energy above 3.8 kWh/km on a flat road with fewer than ten passengers, energy below 0.35 kWh/km without a descent, and SoC at or below 5%. High consumption on steep or crowded trips, and at extreme temperatures, is kept on purpose, since it is genuine behaviour that a generic IQR or z-score filter would delete. Missing values would be imputed with route medians, but none occurred. No record was removed from either dataset; 1,405 steep-gradient (|G| > 2%) and 1,482 extreme-temperature RZ-Syn trips were retained. A retention of 100% says more about how the simulators bound their output than about the filter.

Grouped Splitting

Buses, not trips, are assigned to partitions, so that no bus contributes to both training and testing. RZ-Syn gives 23 training, 4 validation and 6 test buses (seed 42), which is 7,350, 1,245 and 1,864 trips; IN-Syn gives 5,808, 1,243 and 1,449 trips (seed 108). The trip shares (70.3/11.9/17.8% for RZ-Syn) differ from the 70/15/15 bus-level target because buses make unequal numbers of trips. An assertion in the splitter confirms that the partitions share no bus.

METHOD

Feature Engineering

The seven pre-trip inputs (distance, mean gradient, mean passenger count, stops per kilometre, mean speed, departure SoC and ambient temperature) are expanded into 19 features (Table II) that translate the physics into columns a tree or a linear model can use. The square of speed stands for aerodynamic drag. Gradient times distance is a proxy for net elevation change, and gradient times passenger count for the extra gravitational work done on payload. Stop density, speed times stops and distance per stop describe kinetic energy lost to braking. Temperature deviation from a 21 °C comfort setpoint, its square, and heating and cooling degree terms describe the V-shaped HVAC demand. Two features flag the high-resistance regime below 25% SoC. Every feature is computed from quantities known before departure; nothing from the completed trip enters the model, which is what makes the forecast usable for dispatch.

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue X, October 2026

TABLE II

ENGINEERED FEATURE SET (19 COLUMNS)

Mechanism	Features
Base inputs (7)	distance_km, avg_gradient_percent, avg_passenger_count, stops_per_km, avg_speed_kmh, initial_soc_percent, ambient_temp_degc
Aerodynamic and kinetic	speed_squared, speed_x_stops, distance_per_stop
Gravity and payload	gradient_x_distance, gradient_x_passenger, estimated_total_mass_tonnes
Thermal (HVAC)	temp_deviation, temp_deviation_squared, heating_degree_demand, cooling_degree_demand
Battery	low_soc_flag, soc_resistance_proxy

Models and Training

Four regressors are compared. Linear regression on standardised features is the baseline. The others are a random forest [15] (100 trees, depth 12, minimum split 4, minimum leaf 2), histogram gradient boosting, the scikit-learn implementation [16] of the gradient-boosting idea [14] in the style of LightGBM [13] (150 iterations, depth 6, learning rate 0.08), and XGBoost [12] (150 trees, depth 5, learning rate 0.06, row and column subsampling 0.85). All use a fixed random seed (42 for RZ-Syn, 108 for IN-Syn) and train on a standard laptop CPU. The project code selects the model with the lowest test RMSE. Selecting on the test set is not good practice, so we checked the alternative: ranked by validation RMSE the order is the same (histogram gradient boosting 0.0535, XGBoost 0.0540, random forest 0.0662, linear regression 0.1314 kWh/km), and the winner does not change.

C. Metrics

Predictions are scored with the mean absolute error (MAE), the root mean squared error (RMSE), the coefficient of determination (R^2) and the mean absolute percentage error (MAPE). With yi the true rate, ŷi the prediction and n the number of test trips,

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2} \quad (5)$$

$$\mathrm{MAPE} = \frac{100}{n}\sum_{i=1}^{n}\frac{|\hat{y}_i - y_i|}{y_i} \quad (6)$$

MAE is the mean of $|\hat{y}i - yi|$ and R^2 is one minus the ratio of residual to total sum of squares. MAE and RMSE are in kWh/km. Confidence intervals come from 2,000 bootstrap resamples of the test trips, and the model comparison is repeated over 20 re-drawn bus-grouped splits to expose split-to-split variation. To separate model error from irreducible noise we also compute a noise ceiling: the metrics of the noise-free generator equation against the noisy targets of the same test trips. No learner that sees only the model inputs can be expected to beat it.

Explainability

SHAP values [11] attribute each prediction to the input features. For the selected boosted model the pipeline uses SHAP's model-agnostic explainer on a random sample of 300 test trips with a 100-trip background, which yields both global importance (mean absolute SHAP value) and per-trip attributions. As an independent check we compute permutation importance on all test trips (10 shuffles per feature, scored by the increase in mean squared error) and compare the two rankings.

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue X, October 2026

Range Assurance

Given a scheduled trip and the bus's SoC at departure, the forecast ê (kWh/km) is multiplied by the distance d to give expected energy. A worst-case figure uses an upper bound on the rate, ê(1 + δ), with δ = 0.065, which the code describes as a 95% interval (Section VI-F tests that description). With battery capacity C = 300 kWh the worst-case arrival SoC is

$$\mathrm{SoC}_{\mathrm{wc}} = \mathrm{SoC}_0 - 100(1+\delta)\frac{\hat{e}\,d}{C} \qquad (7)$$

A trip is classed SAFE if SoCwc is at least 8 percentage points above the 15% reserve, CAUTION if it lies between the reserve and that margin, and HIGH_RISK_DEPLETION below the reserve. In the last case dispatch is blocked and the module returns the top-up energy (kWh and % SoC) that would restore the reserve.

Interval Calibration

The ±6.5% band used by the range-assurance module is a constant written into the service code. To test it, and to offer a principled replacement, we use split-conformal calibration. For each validation trip we compute the absolute relative error $|\hat{y} - y|/\hat{y}$ of the fixed model. The half-width for coverage $1 - \alpha$ is the $\lceil (n + 1)(1 - \alpha) \rceil$-th smallest of these n values, and the resulting band $\hat{y}(1 \pm \text{half-width})$ is then evaluated on the untouched test trips. Under exchangeability this construction guarantees coverage of at least $1 - \alpha$ on average, with no change to the model.

Depot Charging Scheduler

Overnight, each bus must reach a target SoC that covers its next duty plus the 15% reserve, capped at 95%. Energy is drawn through 120 kW chargers at 92% efficiency under an 800 kW depot limit and a five-band time-of-use tariff ($0.08/kWh from 00:00 to 06:00, $0.16 from 06:00 to 09:00, $0.14 from 09:00 to 17:00, $0.28 from 17:00 to 21:00 and $0.12 from 21:00 to 24:00). The scheduler is a greedy heuristic, not an exact optimiser: for each bus it sorts the hours between arrival and departure by price and then by existing depot load, and fills them in that order until the energy requirement is met. Its baseline is uncontrolled charging, in which every bus draws full charger power from the moment it arrives.

Implementation and Verification

The code base is organised as a Python package (data, features, models, optimisation) with a separate FastAPI backend, Pydantic input validation, and joblib model artefacts. Twenty-three pytest tests cover the loader and schema, outlier rules, feature engineering, group-split leakage, metrics, artefact loading, prediction, the two decision modules and the API; the test cache records all 23 collected and none failed. Beyond the project's own tests, we re-ran the evaluation independently: the saved histogram-gradient-boosting artefact reproduces its reported test metrics exactly, linear regression and random forest reproduce when retrained, and a retrained boosted model differs slightly because the library version differs (Section VI-A). The XGBoost artefact could not be reloaded in our analysis environment, so its figures are those stored with the project.

Reproducibility and Data Availability

Both datasets are generated by the project's simulators from fixed random seeds (42 for RZ-Syn and 108 for IN-Syn), so they can be regenerated exactly. The saved models were trained with scikit-learn 1.9.1 and the independent re-evaluation used version 1.8.0, as noted in Section VI-A. Splits, bootstrap resamples and stress tests use stated seeds (42 and 100–119 for splits, 2,000 bootstrap resamples, 2026 for the range-assurance stress test). The simulators, source code, trained models and test suite are available from the corresponding author on request.

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue X, October 2026

RESULTS AND DISCUSSION

Model Comparison on RZ-Syn

Table III gives the test results on the six held-out buses. The two boosted models are clearly ahead of the random forest and far ahead of the linear baseline, whose MAPE of 11.96% reflects a relationship that is strongly nonlinear, especially around the clip floor. Histogram gradient boosting is best on every metric (MAE 0.0374 kWh/km, RMSE 0.0541 kWh/km, R^2 0.9916, MAPE 3.45%), with XGBoost close behind. Because the XGBoost artefact could not be rerun here, we do not claim a statistically established ordering between those two. Bootstrap 95% intervals for the selected model are narrow: MAE 0.0355–0.0392, RMSE 0.0511–0.0570, R^2 0.9906–0.9925 and MAPE 3.31–3.59%. The paired bootstrap difference in MAE between histogram boosting and the random forest is −0.0075 to −0.0046 kWh/km, an interval that excludes zero, so that gap is not sampling noise.

TABLE III

TEST RESULTS ON RZ-SYN (1,864 TRIPS, 6 HELD-OUT BUSES)

Model	Val RMSE	Test MAE	Test RMSE	Test R^2	MAPE (%)
Linear regression	0.1314	0.0921	0.1248	0.9552	11.96
Random forest	0.0662	0.0434	0.0666	0.9872	3.8
XGBoost	0.054	0.0391	0.0557	0.9911	3.71
Hist. gradient boosting	**0.0535**	**0.0374**	**0.0541**	**0.9916**	**3.45**
Noise ceiling	–	0.0281	0.0421	0.9949	2.41

MAE and RMSE in kWh/km. The noise ceiling scores the noise-free generator against the same noisy targets.

The last row of Table III is the noise ceiling. An R^2 of 0.9916 sits 0.0033 below the 0.9949 that even a perfect model of the physics could reach on this data, and the model's MAE is 0.0093 kWh/km above the ceiling's. The model therefore recovers almost all of the structure the generator put in. That is a statement about the simulator, and it does not carry over automatically to buses on a road.

Retraining with scikit-learn 1.8.0 reproduced linear regression and the random forest to within 0.0001 kWh/km, but the boosted model came out slightly different from the artefact trained with version 1.9.1 (MAE 0.0383, RMSE 0.0552, R^2 0.9912, MAPE 3.52%). We take that as the size of the reproducibility gap across library versions. Fig. 4 shows the saved model's predictions against the truth. They follow the diagonal over the whole range, the residuals are centred (mean −0.0011 kWh/km, SD 0.0541) and 50.4% of trips are under-predicted, so there is no systematic bias. Converted to whole-trip energy, the MAE is 0.54 kWh on a mean trip of 15.6 kWh (3.45%).

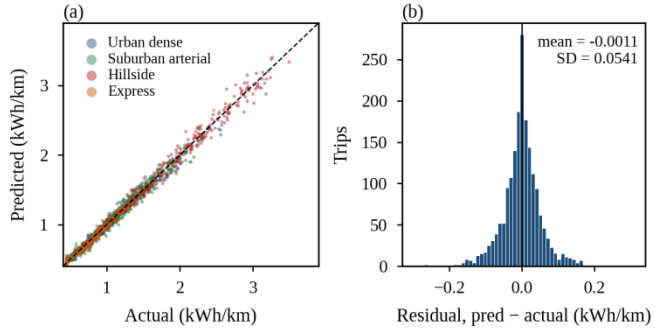


Fig. 4. Held-out test results on RZ-Syn for the selected model. (a) Predicted versus actual rate, coloured by route archetype. (b) Residual distribution (prediction minus actual).

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue X, October 2026

A single split can flatter or punish a model by luck, so the comparison was repeated. The 33 buses were re-drawn into 20 different bus-grouped train, validation and test partitions (seeds 100–119, 1,808–1,983 test trips each), and linear regression, the random forest and histogram boosting were retrained on each (Table IV). XGBoost is omitted because its artefact could not be reloaded here. The ordering never changed: histogram boosting beat the random forest and the linear baseline on MAE in all 20 splits. Its MAE of 0.0367 ± 0.0008 kWh/km (range 0.0354–0.0383), R^2 of 0.9914 ± 0.0005 and MAPE of 3.44 ± 0.08% are close to the headline split, which therefore was not a lucky draw.

TABLE IV

ROBUSTNESS OVER 20 RE-DRAWN BUS-GROUPED SPLITS (RZ-SYN)

Model	Test MAE	Test R^2	MAPE (%)
Linear regression	0.0939 ± 0.0016	0.9478 ± 0.0038	12.72 ± 0.34
Random forest	0.0422 ± 0.0012	0.9873 ± 0.0008	3.77 ± 0.11
Hist. gradient boosting	**0.0367 ± 0.0008**	**0.9914 ± 0.0005**	**3.44 ± 0.08**

Mean ± SD over 20 splits (seeds 100–119); MAE in kWh/km. Histogram boosting had the lowest MAE in every split.

Model Comparison on IN-Syn

The ranking is the same on the Indian benchmark (Table V). Histogram gradient boosting, the selected model, reaches R^2 0.9829, MAE 0.0607 kWh/km and MAPE 3.83%; XGBoost has a marginally lower MAE and MAPE (0.0601 kWh/km, 3.76%) at the same R^2, so the two are effectively tied. Absolute errors are larger than on RZ-Syn partly because mean consumption is higher (1.61 against 1.07 kWh/km); in relative terms the two datasets are close (MAPE 3.8% against 3.5%). The gap to the noise ceiling is wider here, 0.0198 kWh/km in MAE against 0.0093, which is consistent with the hidden road-roughness, AC and bus-size variables that the seven inputs cannot carry. By city, MAPE runs from 3.35% (Hyderabad) to 4.49% (Shimla), and the hill cities have the largest absolute errors (0.103 kWh/km in Shimla and 0.115 in Dehradun–Mussoorie), in line with their higher consumption of 2.4–2.8 kWh/km.

TABLE V

TEST RESULTS ON IN-SYN (1,449 TRIPS, 6 HELD-OUT BUSES)

Model	Val RMSE	Test MAE	Test RMSE	Test R^2	MAPE (%)
Linear regression	0.1448	0.1074	0.1387	0.9504	7.3
Random forest	0.0954	0.0706	0.097	0.9758	4.48
XGBoost	0.0778	0.0601	0.0816	0.9828	3.76
Hist. gradient boosting	**0.0769**	**0.0607**	**0.0814**	**0.9829**	**3.83**
Noise ceiling	–	0.0409	0.0553	0.9921	2.52

The ceiling uses all simulator inputs, including road roughness, AC state and bus size.

What the Engineered Features Contribute

Table VI removes information in two ways. First, replacing the 19 features with the 7 raw inputs hurts linear regression badly (R^2 from 0.955 to 0.859, MAPE from 12.0% to 19.6%), as expected of a model that cannot form products or squares for itself. For the tree-based models the effect is small: boosting moves from R^2 0.9902 to 0.9912 and from MAPE 3.72% to 3.52%, and the random forest from 0.9864 to 0.9872. Those differences are about the size of the bootstrap uncertainty (roughly ±0.002 kWh/km in MAE), so we would not claim that feature engineering improves tree models here; trees can find the interactions from the raw

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue X, October 2026

inputs. Second, dropping one physical group at a time shows where the signal lives. Without the gradient group R^2 collapses to 0.26, without the thermal group it falls to 0.878, and without the speed and stop group or the mass and passenger group it stays above 0.98. Removing the SoC group changes nothing (0.9914 against 0.9912), because SoC below 25% almost never occurs in the data.

TABLE VI

ABLATION ON RZ-SYN TEST SET

Configuration	R^2	MAE	MAPE (%)
Linear regression, 7 raw inputs	0.8587	0.1676	19.56
Linear regression, 19 features	0.9552	0.0921	11.96
Random forest, 7 raw inputs	0.9864	0.0453	4.02
Random forest, 19 features	0.9872	0.0435	3.8
Boosting, 7 raw inputs	0.9902	0.0399	3.72
Boosting, all 19 features	**0.9912**	**0.0383**	**3.52**
Boosting, no gradient group	0.2596	0.3903	44.61
Boosting, no thermal group	0.8783	0.1398	13.75
Boosting, no speed/stop group	0.9822	0.0507	4.58
Boosting, no mass/passenger group	0.9854	0.0492	4.41
Boosting, no SoC group	0.9914	0.038	3.51

Boosted models retrained with scikit-learn 1.8.0, hence 0.9912 rather than the saved artefact's 0.9916. Groups: gradient (3 features), thermal (5), speed and stops (5), mass and passengers (3, sharing gradient × passenger with the gradient group), SoC (3).

Error by Operating Regime and by Route

In relative terms the errors are uniform across the four route archetypes (MAPE 3.2–3.6%, Table VII); in absolute terms they grow with consumption, from 0.028 kWh/km on express routes to 0.057 on hillsides. Mean bias is below 0.0035 kWh/km in every archetype. The project's 42-route report applies the saved model to every trip on each route, training trips included, so it is not a held-out measure. It gives a mean route-level accuracy (100 − MAPE) of 97.2%, with a range of 96.8–97.7%, and route-mean predicted rates that sit within 0.002 kWh/km of the actual means on average. At 85% departure SoC on a 300 kWh pack, the mean trip on every one of the 42 routes is classed SAFE, with an estimated maximum range between 129 km on the steepest hillside route and 357 km on an express route.

TABLE VII

HELD-OUT ERROR BY ROUTE ARCHETYPE (RZ-SYN)

Archetype	Trips	Mean rate	MAE	MAPE (%)	Bias
Urban dense	657	0.954	0.0343	3.49	−0.0034
Suburban arterial	622	0.98	0.0354	3.59	0.001
Hillside	302	1.813	0.0574	3.23	−0.0003
Express	283	0.808	0.0276	3.27	−0.0012

Rates, MAE and bias in kWh/km; bias is prediction minus actual.

Explaining the Forecasts

Road gradient dominates the SHAP ranking (Fig. 5, Table VIII): its mean absolute SHAP value of 0.3104 kWh/km is nearly three times that of the runner-up, temperature deviation (0.1127). The two gradient

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue X, October 2026

interactions follow (0.0814 and 0.0733), then mean speed (0.0299). In the summary plot, steeper positive gradients and larger temperature deviations push the forecast upward, while the SoC, mass and squared-term features sit on the zero line. Permutation importance on the full test set agrees: the same five features in the same order, and a Spearman rank correlation of 0.935 across all 19 features.

Two cautions apply. First, gradient, gradient × distance and gradient × passengers are closely related, so SHAP divides credit among them in ways that depend on the explainer, and speed squared receives almost none because mean speed carries the same information. The mechanism is better read at group level (Table VI) than feature by feature. Second, the ranking mirrors the generator. An empty 12.5-tonne bus on a 2% grade needs about 2.5 kN of climbing force against 1.2 kN of rolling resistance, so gradient dominates by construction. Whether it also dominates on a real fleet is a hypothesis, one that earlier simulation studies support [3], [7], and not something this experiment can establish.

TABLE VIII

GLOBAL IMPORTANCE: SHAP VERSUS PERMUTATION

Feature	SHAP rank	Mean |SHAP|	Perm. ΔMSE
avg_gradient_percent	1	0.3104	0.3592
temp_deviation	2	0.1127	0.0479
gradient_x_distance	3	0.0814	0.0207
gradient_x_passenger	4	0.0733	0.0142
avg_speed_kmh	5	0.0299	0.0079
ambient_temp_degc	6	< 0.03	0.0031
heating_degree_demand	7	< 0.03	0.0012
speed_x_stops	8	< 0.03	0.0003

Mean |SHAP| in kWh/km, from the pipeline output for the top five features; ranks 6–8 are read from the saved bar chart. Rank correlation over all 19 features: ρ = 0.935.

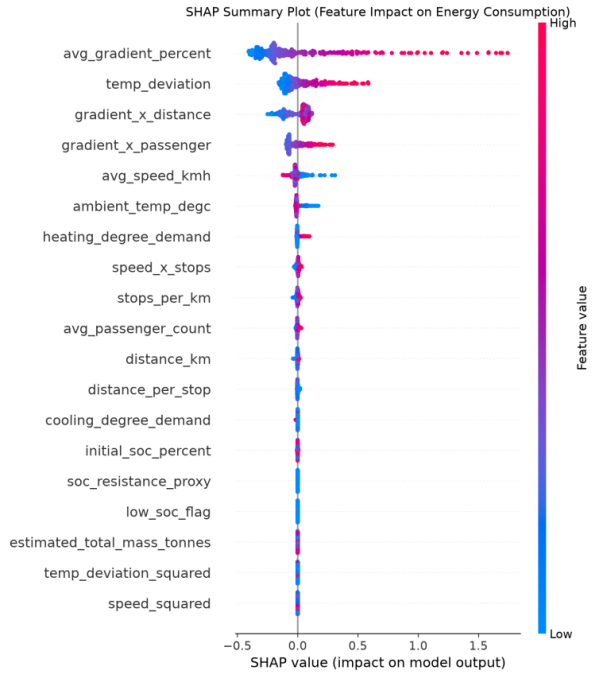


Fig. 5. SHAP summary plot for the selected model on 300 test trips (RZ-Syn). Each dot is a trip; colour shows the feature value.

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue X, October 2026

Calibration of the Prediction Band

The service attaches a ±6.5% band to every forecast and labels it a 95% interval, and the range-assurance module uses its upper edge as the worst case. On the held-out trips the band contains the true rate for 86.1% of trips (86.0% on the validation partition), and 7.5% of trips exceed its upper edge. A half-width of ±7.3% would be needed for 90% coverage and ±9.3% for 95%. The shortfall is of the same kind as the 79.4% against 90% reported by Almajed et al. [10] for their quantile intervals.

The standard remedy, split-conformal calibration, was applied (Section V-F). On the 1,245 validation trips the 95% half-width is ±9.06%, and on the held-out test trips this band covers 94.5% of rates, close to its nominal level; the 90% half-width of ±7.52% covers 91.0%. Calibration is cheap and does not need retraining. Section VI-H shows what it costs in the dispatch decision. The shipped ±6.5% value was left unchanged in the service so that every other result describes the system as delivered.

Generalisation Stress Tests

Fig. 6 and Table IX put the headline figure in context. The first three rows are in-distribution: the reported test, five-fold validation grouped by bus, and six-fold validation grouped by route all give essentially the same MAPE (3.37–3.45%) with tiny spread. That is reassuring but uninformative, since in the simulator neither bus nor route identity affects energy; a trip's features tell the whole story. Grouped splitting prevents leakage, but here it cannot reveal a lack of generalisation because there is nothing group-specific to generalise to.

The harder tests are the ones that remove a whole operating regime. Withholding the hillside archetype raises MAPE to 6.43% and the MAE nearly fourfold to 0.150 kWh/km, because tree ensembles cannot extrapolate beyond the gradients they have seen; the other three archetypes degrade mildly (3.66–4.85%). Moving between benchmarks is far worse. A model trained on RZ-Syn and applied to IN-Syn reaches R^2 0.52 and MAPE 24.2%, with a mean error of −0.31 kWh/km: it under-predicts, which is the dangerous direction for stranding. Many Indian inputs lie outside the RZ-Syn range (16.5% of trips for stops per km, 9.7% for passengers) and the physics differs. The reverse transfer gives R^2 0.80 and MAPE 20.3%. A model has to be trained on the fleet and climate it will serve, and it needs a check for inputs outside its training range before it is trusted.

An obvious safeguard is to flag any trip whose inputs fall outside the training range, so this was tested. For the RZ-Syn model applied to IN-Syn, a per-variable minimum and maximum check on the seven inputs flags 33.9% of trips. Flagged trips are indeed worse (MAE 0.449 against 0.336 kWh/km), but the unflagged trips are still poor (MAPE 23.0%, and 88.9% of them under-predicted). For the withheld hillside archetype the check flags none of the 1,706 trips, because every hillside input value lies inside the range spanned by the other archetypes, yet the MAE is nearly four times larger. A per-variable range check is therefore necessary but not sufficient: the failure comes from combinations of inputs, and catching it would need a multivariate novelty detector.

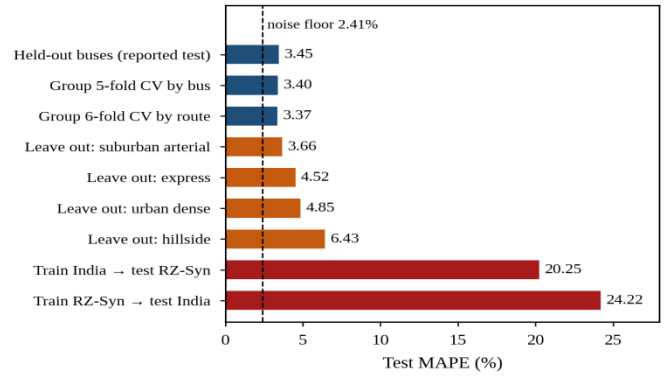


Fig. 6. Test MAPE under progressively harder evaluation regimes. Blue: in-distribution; orange: one route archetype withheld; red: transfer between benchmarks. The dashed line is the RZ-Syn noise ceiling.

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue X, October 2026

TABLE IX

GENERALISATION STRESS TESTS (HISTOGRAM BOOSTING)

Evaluation	MAE	R^2	MAPE (%)
Held-out buses (reported)	0.0374	0.9916	3.45
Group 5-fold by bus	0.0364 ± 0.0010	0.9916 ± 0.0006	3.40 ± 0.08
Group 6-fold by route	0.0361 ± 0.0012	0.9918 ± 0.0008	3.37 ± 0.04
Withhold urban dense	0.0492	0.9602	4.85
Withhold suburban arterial	0.0353	0.9879	3.66
Withhold hillside	0.1503	0.8789	6.43
Withhold express	0.0368	0.979	4.52
Train RZ-Syn, test IN-Syn	0.3774	0.5202	24.22
Train IN-Syn, test RZ-Syn	0.1927	0.8005	20.25

Mean ± SD over folds. Withheld-archetype models are trained on the other three archetypes and tested on the one withheld. MAE in kWh/km.

Range Assurance

On the generous SoC levels in the data, almost every trip is trivially feasible, which makes a poor test. We therefore built a stress test. For each of the 1,864 held-out trips, the departure SoC was set so that the true arrival SoC lay between 8 points below and 12 points above the 15% reserve, uniformly at random, which makes 735 of the trips truly infeasible. Three rules were then compared as detectors of infeasibility (Table X): the system's worst-case rule, a point-forecast rule using the expected arrival SoC, and the static rule that multiplies distance by the fleet-average training rate (1.07 kWh/km).

The worst-case rule caught every infeasible trip (recall 100%) at the cost of 27 false alarms among 1,129 feasible trips (2.4%). The point-forecast rule missed 14 trips (1.9%) with 7 false alarms. The static average rule was both blind and noisy: it missed 76 infeasible trips (10.3%) and raised 116 false alarms (10.3%). Replacing the shipped ±6.5% band by the calibrated ±9.06% band keeps recall at 100% and raises false alarms from 27 to 42 (3.7% of feasible trips), so honest coverage costs about 15 extra unnecessary blocks in 1,129. For the 762 trips blocked by the system, the recommended top-up (12.2 kWh on average) would have restored the reserve in 94.5% of cases. This is the practical case for a trip-level forecast: a static average cannot tell the steep, crowded trip from the flat, empty one. The test is, however, built from the same simulator that trained the model and the stress margins were chosen by us, so it shows the logic behaves as intended and says little about field performance.

TABLE X

DETECTION OF INFEASIBLE TRIPS (735 OF 1,864)

Rule	Missed	Missed (%)	False alarms	Precision (%)
Static average rate	76	10.3	116	85
Point forecast	14	1.9	7	99
Worst case (system)	**0**	**0**	**27**	**96.5**
Worst case, calibrated ±9.06%	0	0	42	94.6

False alarms are counted among the 1,129 trips that are truly feasible. The calibrated row replaces the ±6.5%

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue X, October 2026

band by the split-conformal 95% band of Section VI-F.

Depot Charging

We simulated overnight depots of 5, 10, 20 and 33 buses, each needing the predicted energy of eight randomly drawn trips (about 122 kWh on average) and arriving at 17:00, 19:00, 21:00, 22:00 or 23:00 with a 06:00 departure, which gives 20 scenarios. Table XI summarises them. The scheduler never exceeded the 800 kW limit (maximum 421 kW), whereas uncontrolled charging would have exceeded it in 10 of the 20 scenarios and peaked at 2,264 kW. Fig. 7 shows the 20-bus, 22:00 case: uncontrolled charging produces a 1,096 kW spike in the first hour, while the smart schedule spreads the load at roughly 160–200 kW through the cheap hours from 00:00 to 06:00 and lowers cost from $135.37 to $90.25.

Cost savings ranged from 32.4% to 71.4%, with a mean of 48.5%, and are set almost entirely by the tariff. For arrivals at 17:00 or 19:00 the baseline pays $0.28/kWh against $0.08 for the cheap window (a 71.4% saving), and for arrivals from 21:00 the baseline pays $0.12 (a 33.3% saving). Fleet size made no difference. The saving is therefore a property of the tariff gap and the choice of baseline, not evidence of solver sophistication, and it should not be quoted as a general range. The scheduler left a shortfall below 0.7 kWh fleet-wide in the worst scenario, a consequence of its 5 kW minimum slot, which is negligible. Because it is a greedy heuristic, it can be compared with a mixed-integer formulation for optimality, which we have not done.

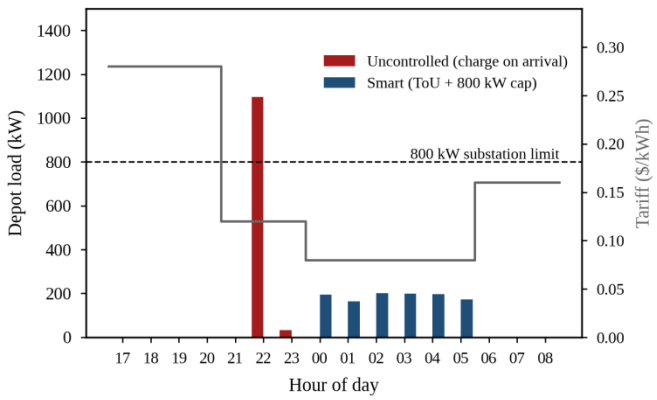


Fig. 7. Depot load for 20 buses arriving at 22:00 and leaving at 06:00. Bars are hourly load; the line is the tariff.

TABLE XI

DEPOT CHARGING SIMULATION (20 SCENARIOS)

Arrival	Saving (%)	Peak uncontrolled (kW)	Peak smart (kW)
17:00	71.4	2,144	392
19:00	71.4	2,182	415
21:00	33.3	2,264	421
22:00	33.3	1,719	304
23:00	33	1,782	317

Mean saving, and maximum peak, over fleets of 5, 10, 20 and 33 buses. Departure 06:00; 800 kW depot limit.

Runtime

A single forecast through the Python interface, including data-frame construction and feature engineering, takes 8.2 ms on average on our analysis machine, and a batch of 1,864 trips takes 27 ms, excluding network

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue X, October 2026

overhead. The selected model's artefact is 495 kB. This is comfortably within what a depot scheduling system needs.

DISCUSSION

Taken together, the results support a modest and specific claim. On data in which energy is a smooth physical function of seven pre-trip variables plus a few percent of noise, a boosted-tree model recovers that function almost to the noise ceiling, its explanations agree with the physics that generated the data, and a forecast-driven range check is much better than a static average at spotting infeasible trips. The ranking of models is stable across 20 re-drawn bus splits. The headline MAPE of 3.45% is of the same order as the 3.51% reported for a parametric tropical dataset [10], but neither figure is evidence about real buses, and we caution against comparing them with MAPEs measured on telemetry.

The diagnostic results matter more than the headline. The model degrades sharply, and in the unsafe direction, when the operating regime changes; simple range checks do not catch this; and the interval that ships with the service is overconfident but can be repaired cheaply.

Threats to Validity

Construct validity. (1) All data are synthetic, so accuracy is partly circular: the model learns the equations that made the targets, and real telemetry adds sensor noise, driver behaviour, traffic and battery ageing that no benchmark here contains. (2) The generators have artefacts: a clip floor on 16.1% of RZ-Syn trips, almost no low-SoC trips, and two-valued SoC in IN-Syn.

Internal validity. (3) The project code selects the model on test RMSE. The validation ranking agrees, but a clean protocol would select on validation data only. (4) The repeated-split study covers three of the four models and uses one learner seed. The XGBoost artefact could not be reloaded in our environment, and SHAP values are taken from the saved pipeline output rather than recomputed.

External validity. (5) The grouped split is sound practice but untestable here, as shown in Section VI-G. Performance falls under regime and benchmark shift, and the per-variable range check does not reliably warn of it.

Decision validity. (6) The shipped interval undercovers; the calibrated band repairs coverage but was not adopted in the service. (7) The depot scheduler is a heuristic whose reported savings follow from the tariff and the baseline. None of these overturns the pipeline's design, but each limits what may be claimed from it.

CONCLUSION AND FUTURE WORK

We built and evaluated an end-to-end pipeline for trip-level BEB energy forecasting: physically motivated features, four regressors trained under a bus-grouped split, SHAP explanations, and two decision modules that turn a forecast into a dispatch decision and an overnight charging plan, all behind a REST service. On the two synthetic benchmarks, histogram gradient boosting reached R^2 0.9916 (MAPE 3.45%) on RouteZero-style data, with a MAPE of 3.44 ± 0.08% over 20 re-drawn bus splits, and R^2 0.9829 (MAPE 3.83%) on Indian-conditions data, close to the noise ceilings of the generators. In a low-battery dispatch test the forecast-driven worst-case rule found every infeasible trip, where a static average missed one in ten.

The stress tests are as much a result as the headline figures. Accuracy fell when a whole operating regime was withheld and fell sharply, with under-prediction, when a model crossed from one benchmark to the other, and a per-variable input-range check did not reliably warn of either. The nominal 95% interval covered 86.1% of trips; split-conformal recalibration restored 94.5% at the price of about 15 extra unnecessary blocks in 1,129 feasible trips. The quoted charging savings are a property of the tariff, and the generators contain artefacts that limit what the data can show. What the study demonstrates is that the pipeline works and behaves sensibly on data with known structure. It does not demonstrate accuracy on real buses.

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue X, October 2026

Six steps would turn this into a deployable tool. First, fit and validate the models on measured trip telemetry from an operating fleet, and recalibrate the simulators against it. Second, adopt the conformal band in the service in place of the fixed ±6.5%, recalibrating it on fleet data. Third, add a multivariate novelty detector, since the per-variable range check tested here is not enough to flag out-of-distribution trips. Fourth, supply road roughness, AC state and bus size as model inputs for Indian operation, repair the two-valued SoC in IN-Syn, and generate enough low-SoC trips to exercise the battery mechanism. Fifth, select models on validation data only, extend the repeated-split study to XGBoost, and compare the greedy scheduler with an exact mixed-integer formulation that includes demand charges and battery ageing, the later phases of our original plan. Sixth, log actual against predicted consumption in service and retrain periodically.

ACKNOWLEDGMENT

The authors thank their project guide, Dr. Khasim Syed, for his guidance throughout the CAP4001 capstone project.

REFERENCES

1. B. Sturmberg, J. N. Hendriks, et al., “An integrated model of electric bus energy consumption and optimised depot charging,” npj Sustain. Mobil. Transp., 2024, doi: 10.1038/s44333-024-00008-2.
2. H. Basma, C. Mansour, M. Haddad, M. Nemer, and P. Stabat, “Comprehensive energy modeling methodology for battery electric buses,” Energy, vol. 207, Art. no. 118241, 2020.
3. H. Abdelaty, A. Al-Obaidi, M. Mohamed, and H. E. Z. Farag, “Machine learning prediction models for battery-electric bus energy consumption in transit,” Transp. Res. D, Transp. Environ., vol. 96, Art. no. 102868, 2021.
4. H. Abdelaty and M. Mohamed, “A prediction model for battery electric bus energy consumption in transit,” Energies, vol. 14, no. 10, Art. no. 2824, 2021.
5. S. Nan, R. Tu, T. Li, J. Sun, and H. Chen, “From driving behavior to energy consumption: A novel method to predict the energy consumption of electric bus,” Energy, vol. 261, Art. no. 125188, 2022.
6. S. Nan, F. Liao, T. Li, H. Chen, and J. Sun, “Identifying the electricity-saving driving behaviors of electric bus based on trip-level electricity consumption: A machine learning approach,” Environ. Sci. Pollut. Res., vol. 30, no. 34, pp. 82743–82759, 2023.
7. W. Wang, M. Mohamed, B. Zhang, Q. Zhang, and H. Abdelaty, “Electric bus energy prediction and factors interactions using explainable machine learning models,” Eng. Appl. Artif. Intell., vol. 163, Art. no. 112877, 2026.
8. J. Vepsäläinen, A. Ritari, A. Lajunen, K. Kivekäs, and K. Tammi, “Energy uncertainty analysis of electric buses,” Energies, vol. 11, no. 12, Art. no. 3267, 2018.
9. M. Gallet, T. Massier, and T. Hamacher, “Estimation of the energy demand of electric buses based on real-world data for large-scale public transport networks,” Appl. Energy, vol. 230, pp. 344–356, 2018.
10. M. Almajed, M. Hassan, T. D. Turjo, M. A. Islam, M. E. Haque, and M. M. H. Rahman, “Explainable machine learning for electric bus energy prediction under tropical urban conditions: A physics-informed parametric framework,” Energy Eng., vol. 123, no. 10, 2026, doi: 10.32604/ee.2026.084024.
11. S. M. Lundberg and S.-I. Lee, “A unified approach to interpreting model predictions,” in Proc. Adv. Neural Inf. Process. Syst., vol. 30, 2017, pp. 4765–4774.
12. T. Chen and C. Guestrin, “XGBoost: A scalable tree boosting system,” in Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, 2016, pp. 785–794.
13. G. Ke et al., “LightGBM: A highly efficient gradient boosting decision tree,” in Proc. Adv. Neural Inf. Process. Syst., vol. 30, 2017, pp. 3149–3157.
14. J. H. Friedman, “Greedy function approximation: A gradient boosting machine,” Ann. Statist., vol. 29, no. 5, pp. 1189–1232, 2001.
15. L. Breiman, “Random forests,” Mach. Learn., vol. 45, no. 1, pp. 5–32, 2001.
16. F. Pedregosa et al., “Scikit-learn: Machine learning in Python,” J. Mach. Learn. Res., vol. 12, pp. 2825–2830, 2011.
17. D. R. Roberts et al., “Cross-validation strategies for data with temporal, spatial, hierarchical, or phylogenetic structure,” Ecography, vol. 40, no. 8, pp. 913–929, 2017.

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue X, October 2026

18. S. Kaufman, S. Rosset, C. Perlich, and O. Stitelman, “Leakage in data mining: Formulation, detection, and avoidance,” ACM Trans. Knowl. Discovery Data, vol. 6, no. 4, Art. no. 15, 2012.
19. A. Lajunen, “Energy consumption and cost-benefit analysis of hybrid and electric city buses,” Transp. Res. C, Emerg. Technol., vol. 38, pp. 1–15, 2014.
20. K. R. Kambly and T. H. Bradley, “Estimating the HVAC energy consumption of plug-in electric vehicles,” J. Power Sources, vol. 259, pp. 117–124, 2014.
21. Road Vehicles—Vehicle to Grid Communication Interface—Part 1: General Information and Use-Case Definition, ISO Standard 15118-1, 2019.