Erciyes University Journal of Institute Of Science and Technology

Research Article
10.65520/erciyesfen.1968962

Imprint:
Volume: 42(3)
Year: 2026
Page: 467-493

Ramazan ENİSOĞLU[a*]
Burak Alp İNAN[b]

[a] Dr., Kırıkkale University, ramazanenisoglu@kku.edu.tr
[b] Dr., Hacettepe University, inan@ee.hacettepe.edu.tr

* Corresponding Author

Received: 6/11/2026
Accepted: 10/19/2026

Citation:
Ramazan ENİSOĞLU, Burak Alp İNAN (2026). Energy Consumption Prediction for 5G Base Stations: A Comparative Machine Learning Study under Hardware Heterogeneity. *Erciyes University Journal of Institute Of Science and Technology*, 42(3), 467-493. https://doi.org/10.65520/erciyesfen.*1968962*

Screened by iThenticate for Authors & Researchers

Except where otherwise noted, content in this article is licensed under a Creative Commons 4.0 International license. Icons by Font Awesome.

Energy Consumption Prediction for 5G Base Stations: A Comparative Machine Learning Study under Hardware Heterogeneity

Abstract

Background/Aim: Energy consumption of 5G base stations has become a central operational concern, and energy-saving mechanisms such as carrier shutdown and sleep modes require accurate prediction of station-level consumption. This study compares prediction models under conditions reflecting deployment to unseen base stations.

Methods: Seven models – from a trivial baseline and linear models to tree ensembles and a multilayer perceptron – were evaluated on the ITU 5G Base Station Energy Consumption Modelling dataset (98,084 hourly observations from 923 base stations with heterogeneous hardware) using station-grouped five-fold cross-validation, station-clustered bootstrap intervals, repeated station partitions and out-of-time tests. An analytical load-dependent power model and four published data-driven approaches were re-implemented under the identical protocol.

Results: LightGBM attained the best accuracy among the seven models (R^2 = 0.910, WMAPE = 10.22%), matched in R^2 by the multilayer perceptron at roughly twenty times the training cost. LightGBM remained within 0.15 percentage points of the most accurate re-implemented technique, whereas the analytical model was clearly less accurate; lower published errors (WMAPE of about 4–5%) arise only when test stations were seen during training. Consumption was governed predominantly by static hardware attributes and their interaction with traffic load. Accuracy was stable across the observed days, but tree ensembles underestimated consumption when load exceeded the training range.

Conclusion: Gradient boosting offers the most favourable accuracy–cost trade-off for unseen base stations, while seasonal use requires periodic recalibration. The study also documents a sparsity-induced instability in linear models and the hardware distribution shift between training and evaluation data.

Keywords: 5G, Energy Consumption, Machine Learning, Base Station, Gradient Boosting

5G Baz İstasyonları için Enerji Tüketimi Tahmini: Donanım Heterojenliği Altında Karşılaştırmalı Makine Öğrenimi Çalışması

Öz

5G baz istasyonlarının enerji tüketimi temel bir işletim sorunu hâline gelmiştir ve taşıyıcı kapatma ile uyku modları gibi enerji tasarrufu mekanizmaları, istasyon düzeyinde tüketimin doğru tahmin edilmesini gerektirmektedir. Bu çalışmada, basit bir temel modelden doğrusal modellere, ağaç tabanlı topluluk yöntemlerinden çok katmanlı algılayıcıya kadar yedi model, donanım açısından heterojen 923 baz istasyonundan elde edilen 98.084 saatlik gözlemden oluşan

ITU 5G Baz İstasyonu Enerji Tüketimi Modelleme veri kümesi üzerinde karşılaştırılmıştır. Değerlendirmede istasyon gruplandırmalı beş katlı çapraz doğrulama, istasyon kümeli bootstrap güven aralıkları, farklı istasyon bölmeleri ve zaman dışı testler kullanılmış; analitik yük bağımlı bir güç modeli ile literatürdeki dört veri odaklı yaklaşım aynı protokol altında yeniden uygulanmıştır. LightGBM yedi model arasında en iyi doğruluğa ulaşmış (R^2 = 0.910, WMAPE = %10.22), çok katmanlı algılayıcı ise R^2 açısından aynı seviyeyi yaklaşık yirmi kat daha yüksek eğitim maliyetiyle yakalamıştır. LightGBM, yeniden uygulanan en doğru tekniğe 0.15 yüzde puan mesafede kalmış, analitik model ise belirgin biçimde daha az doğru bulunmuştur; literatürdeki daha düşük hatalar (WMAPE yaklaşık %4–5) yalnızca test istasyonları eğitimde görüldüğünde ortaya çıkmaktadır. Enerji tüketimi ağırlıklı olarak sabit donanım özellikleri ve bunların trafik yüküyle etkileşimi tarafından belirlenmektedir. Doğruluk gözlenen günler boyunca kararlı kalmış, ancak yük eğitim aralığını aştığında ağaç tabanlı modeller tüketimi düşük tahmin etmiştir. Sonuç olarak gradient boosting, görülmemiş baz istasyonları için en uygun doğruluk–maliyet dengesini sunmakta, mevsimler boyunca kullanım ise periyodik yeniden kalibrasyon gerektirmektedir. Çalışma ayrıca doğrusal modellerde seyreklik kaynaklı kararsızlığı ve eğitim ile değerlendirme verileri arasındaki donanım dağılımı değişimini ortaya koymaktadır.

Anahtar kelimeler: 5G, Enerji Tüketimi, Makine Öğrenmesi, Baz İstasyonu, Gradient Boosting

1. INTRODUCTION

The fifth generation (5G) of mobile networks delivers substantial gains in capacity, connectivity, and latency, but these gains are accompanied by a marked increase in energy consumption. Reported estimates indicate that 5G consumes approximately four times the energy of 4G, and that the base stations and the access network account for on the order of 80% of the total energy consumed by a mobile network [1, 20]. As operators continue to densify their deployments, the energy bill has become both an economic and an environmental concern, motivating intensive work on network energy-saving solutions.

The Third Generation Partnership Project (3GPP) has progressively introduced energy-saving mechanisms in its New Radio (NR) specifications, including autonomous cell switch-off and reactivation, and has more recently treated network energy saving as a machine learning use case [2]. The effectiveness of such mechanisms – for example, dynamic carrier shutdown or capacity-based sleep modes – depends critically on the ability to predict, accurately and in advance, how much energy a base station will consume under a given configuration and traffic condition. Constructing such a predictor is challenging: energy consumption depends on many heterogeneous hardware attributes, on stochastic traffic, and on the interaction between the two, and operators increasingly require models that generalize to base-station products and configurations not present in the training data [3, 4].

Machine learning has emerged as the dominant approach to this problem. To foster progress, the International Telecommunication Union (ITU), in collaboration with Huawei, organized the 5G Base Station Energy Consumption Modelling Challenge, releasing a real-world dataset and explicitly framing the task in terms of generalization across base station products and configurations [3]. Subsequent work on this dataset has shown the importance of capturing per-station characteristics: the second-ranked solution attributes its performance to an explicit "energy fingerprint" that resolves the one-to-many mapping between identical input features and differing outputs [4]. Beyond this challenge, data-driven energy and load models such as ECO6G have demonstrated significant operating-expenditure savings when traffic load is forecast accurately [5], and ensemble approaches combining gradient boosting with neural networks have been applied to 5G energy estimation [6]. A hybrid of ridge regression and XGBoost with a mixture-of-experts gate has also been proposed for the same challenge [21].

Despite this activity, a systematic and transparent comparison of model families on this dataset – under an evaluation protocol that explicitly prevents information leakage between training and test base stations – remains limited. In addition, the results published for this dataset were obtained under heterogeneous protocols – record-level cross-validation [6], an official split containing both seen and unseen stations [4] and the hidden challenge test set [21] – so that the reported errors cannot be

CC: BY-NC-ND

compared directly; protocols in which records of the same entity appear in both training and test data are a well-documented source of over-optimistic conclusions [22]. Moreover, while tabular-data research suggests that tree-based models frequently match or exceed deep neural networks on medium-sized datasets [7], this question has not been carefully examined for 5G base-station energy prediction. The present study addresses these gaps. Its contributions are as follows:

• A systematic comparison of seven prediction models on the ITU 5G energy dataset, evaluated under a station-grouped cross-validation protocol that estimates generalization to unseen base stations.

• An analysis of feature importance showing that static hardware attributes and their interaction with traffic load, rather than traffic alone, are the dominant predictors of energy consumption.

• A methodological finding regarding the instability of linear models in the presence of sparsely activated energy-saving signals, together with a simple and effective remedy.

• A documented characterization of the hardware distribution shift between the training and evaluation partitions, identified as a key limitation and a direction for future work.

• A protocol-controlled comparison with existing techniques – an analytical load-dependent power model and four data-driven approaches from the literature – re-implemented under the identical validation protocol, which also reconciles the differing error levels reported for this dataset.

• A set of robustness analyses – a leakage benchmark, repeated station partitions, out-of-time validation and a traffic-shift stress test – that clarify the validity of the reported results and their transferability over time.

The remainder of this paper is organized as follows. Section 2 reviews related work on 5G energy modelling and machine learning for tabular data. Section 3 describes the dataset, feature engineering, models, and evaluation protocol. Section 4 presents the results, and Section 5 discusses their implications and limitations. Section 6 concludes the paper.

2. RELATED WORK

The literature relevant to this study spans three strands: analytical and data-driven modelling of 5G energy consumption, machine-learning approaches to base-station energy and load prediction, and the broader question of model selection for tabular data.

2.1. Energy Consumption Modelling in 5G Networks

The energy efficiency of mobile networks has been studied extensively, as the access network and base stations dominate the overall energy budget [1]. Early modelling efforts were predominantly analytical, expressing base-station power as a function of load, transmission power, and hardware parameters. The widely used EARTH model expresses the input power of a base station as a load-independent component plus a term proportional to the radiated power, with a reduced level in sleep mode [23]; it has since been extended to multi-antenna and multi-carrier architectures and to energy-saving features [24, 25]. Measurements on operational sites confirm an approximately linear dependence of base-station power on traffic load [26], and show that air conditioning of equipment shelters can account for a substantial share of site consumption, which ties the energy use of a site to ambient conditions [27]. A comprehensive survey of such power-consumption models and energy-efficiency metrics is provided in [2, 20], which also reviews the energy-saving solutions standardized by 3GPP across successive New Radio releases, including autonomous cell switch-off and reactivation, and notes that network energy saving has been adopted as a machine-learning use case in recent study items. A recurring conclusion of this line of work is that the benefits of standardized energy-saving mechanisms are insufficient on their own, and that accurate predictive models of network power consumption are required to evaluate and to drive more dynamic solutions [2]. However, purely analytical models struggle to capture the full heterogeneity of real deployments, where energy consumption depends on a large number of interacting hardware and operational factors.

CC: BY-NC-ND

2.2. Machine Learning for Base-Station Energy and Load Prediction

To address this complexity, data-driven approaches have gained prominence. The ECO6G study [5] forecasts traffic load from 52 weeks of measurements collected on a real operational 5G base station and uses the predicted load to estimate energy efficiency and operating-expenditure savings, reporting substantial cost advantages of the learned model over traditional time-series baselines across varying load scenarios. To stimulate broad progress on the problem, the International Telecommunication Union, in collaboration with Huawei, released a real-world dataset within the 5G Base Station Energy Consumption Modelling Challenge, framing the task explicitly in terms of generalization across base-station products and configurations [3]. The evaluation deliberately assigns greater weight to samples corresponding to unseen devices and configurations, emphasizing cross-equipment and cross-configuration generalization rather than in-distribution fitting [3].

Solutions developed on this dataset have highlighted the importance of per-station characteristics. The second-ranked solution in [4] introduces an explicit energy fingerprint, incorporating a base-station identifier to resolve the one-to-many problem in which identical input features map to differing energy outcomes; the authors argue that omitting such station-specific information yields an incomplete feature space and degrades prediction accuracy. Notably, this solution ranked second among more than two hundred competing teams worldwide [4], underlining both the difficulty of the task and the value of the dataset as a benchmark. Ensemble methods have also been applied to the problem: a recent study combines XGBoost, CatBoost and an artificial neural network through a fixed weighted average (weights 0.3, 0.3 and 0.4) and reports a MAPE of 3.56% under record-level five-fold cross-validation [6], and a hybrid in which XGBoost is fitted to the residuals of a ridge regression, combined with a mixture-of-experts gate that routes test samples according to their similarity to the training data, reaches a weighted WMAPE of 6.99% on the hidden challenge test set [21]. Collectively, these works establish that machine learning, and tree-based ensembles in particular, are well-suited to base-station energy modelling, but they typically report the performance of a single proposed method rather than a controlled comparison across model families under a leakage-free evaluation protocol. Because each study adopts a different evaluation protocol, their reported errors are not directly comparable; Section 4.6 therefore re-implements these approaches under a common protocol.

2.3. Model Selection for Tabular Data

The choice between tree-based ensembles and deep neural networks for structured, tabular problems remains an active question. Random forests [8] and gradient-boosted decision trees [9], realized in widely used implementations such as XGBoost [10], CatBoost [28] and LightGBM [11], are standard high-performing baselines for tabular regression. Motivated by the success of deep learning in other domains, several specialized neural architectures have been proposed for tabular data, including the attention-based TabNet [12], transformer-style models [13], and prior-fitted transformer approaches such as TabPFN [19]. Nevertheless, large-scale empirical comparisons have repeatedly found that such models struggle to surpass well-tuned tree ensembles on typical tabular problems. The benchmark study of [7] demonstrates that tree-based models remain state of the art on medium-sized tabular datasets (on the order of ten thousand samples), even before accounting for their considerably lower training cost, and attributes this advantage to inductive biases better matched to the heterogeneous, non-smooth structure of tabular data; the systematic evaluation of [13] reaches a similar conclusion, with gradient-boosted trees remaining a strong baseline. Shwartz-Ziv and Armon likewise found that XGBoost outperformed recent deep tabular models across datasets, while an ensemble of deep models and XGBoost performed better than XGBoost alone [29]. The base-station energy dataset considered here exhibits precisely these characteristics – heterogeneous hardware categories, interacting numerical attributes, and a bimodal target – making the comparison between tree-based and neural models directly relevant.

2.4. Positioning of the Present Work

Building on these three strands, the present study contributes a systematic, transparent comparison of seven models on the ITU 5G energy dataset under a station-grouped cross-validation

CC: BY-NC-ND

protocol that prevents information leakage between training and evaluation base stations. In contrast to single-method studies, it reports the full spectrum from a trivial baseline to gradient boosting and a neural network, accompanied by feature-importance and prediction-quality analyses, and it explicitly documents both a sparsity-induced instability in linear models and the hardware distribution shift between training and evaluation data. It further re-implements existing techniques under the same protocol, so that the study can be positioned quantitatively relative to prior work, and it examines the temporal robustness of the results.

3. MATERIALS AND METHODS

3.1. Dataset

This study employs the publicly available 5G base station energy consumption dataset released within the scope of the ITU Artificial Intelligence/Machine Learning in 5G Challenge. The dataset contains hourly records collected over a one-week period (1–8 January 2023) from a heterogeneous population of base stations (BSs) operating under diverse hardware configurations. The data are distributed across five comma-separated files, summarized in Table 1: a static configuration file (BSinfo), an hourly cell-load file (CLdata), an hourly energy-consumption file (ECdata), and two label-free files reserved for the challenge evaluation (estimation and prediction). The labelled training set is obtained by joining the energy-consumption records (ECdata) with the corresponding cell-load records (CLdata) on the (Time, BS) key, and then attaching the static hardware attributes from BSinfo; for stations with several cells, the attributes of the primary cell (Cell0) were used, since the secondary cells are almost always inactive [4]. The resulting set comprises 98,084 hourly observations originating from 923 distinct base stations, with no missing values. Of these stations, 816 are observed on seven consecutive days and 107 on a single day (2 January). The timestamps should be interpreted with caution: the challenge documentation states only that the data were collected on different days [3], and another study describes the same collection period as 1–8 July 2023 [6], so the season of collection cannot be established with certainty. The exact snapshot of the data used here is identified by a persistent identifier [30] (see Code and Data Availability).

Table 1. Structure of the dataset files.

File	Content	Records
BSinfo	Static BS hardware configuration	1,020
CLdata	Hourly cell load and ES-mode signals	125,575
ECdata	Hourly energy consumption (target)	92,629
estimation	Label-free evaluation set	26,139
prediction	Label-free evaluation set	26,705

Each observation is characterized by two groups of predictors. The first group consists of static hardware attributes: the radio unit type (RUType, twelve categories), the operating mode (Mode), the carrier frequency, the channel bandwidth, the number of antennas, and the maximum transmission power. The second group consists of time-varying operational variables: the normalized traffic load and six energy-saving mode activation signals (ESMode1–ESMode6). The prediction target is the hourly energy consumption of the base station. All numerical fields in the dataset are provided in normalized form by the challenge organizers; absolute physical units are not disclosed, and the energy values are expressed on a normalized scale, with observed values ranging from 0.75 to 100. Table 2 lists the variables together with their types and observed ranges in the training set.

CC: BY-NC-ND

Table 2. Predictor and target variables in the training set.

Variable	Type	Range / Categories
RUType	Categorical (static)	12 types (Type1–Type12)
Mode	Categorical (static)	Mode1, Mode2
Frequency	Numerical (static)	155.6–980.0
Bandwidth	Numerical (static)	2–20
Antennas	Numerical (static)	1, 2, 4, 8, 32, 64
TXpower	Numerical (static)	continuous
load	Numerical (dynamic)	0–1
ESMode1–6	Numerical (dynamic)	0–1 (activation ratio)
Energy	Numerical (target)	0.75–100

3.2. Exploratory Data Analysis

Prior to modelling, the training set was examined to characterise the target distribution, the relationship between traffic and energy, and the degree of hardware heterogeneity across base stations. Figure 1 summarises four key observations. The target variable exhibits a right-skewed, bimodal distribution (Figure 1a), with a primary mode near a normalised value of 18 and a secondary mode near 35–40, suggesting a mixture of low- and high-capacity base-station populations. The relationship between traffic load and energy consumption (Figure 1b) is positive but highly dispersed: at any given load level the energy consumption spans a wide range, indicating that traffic alone is an insufficient predictor and that hardware configuration plays a comparable role.

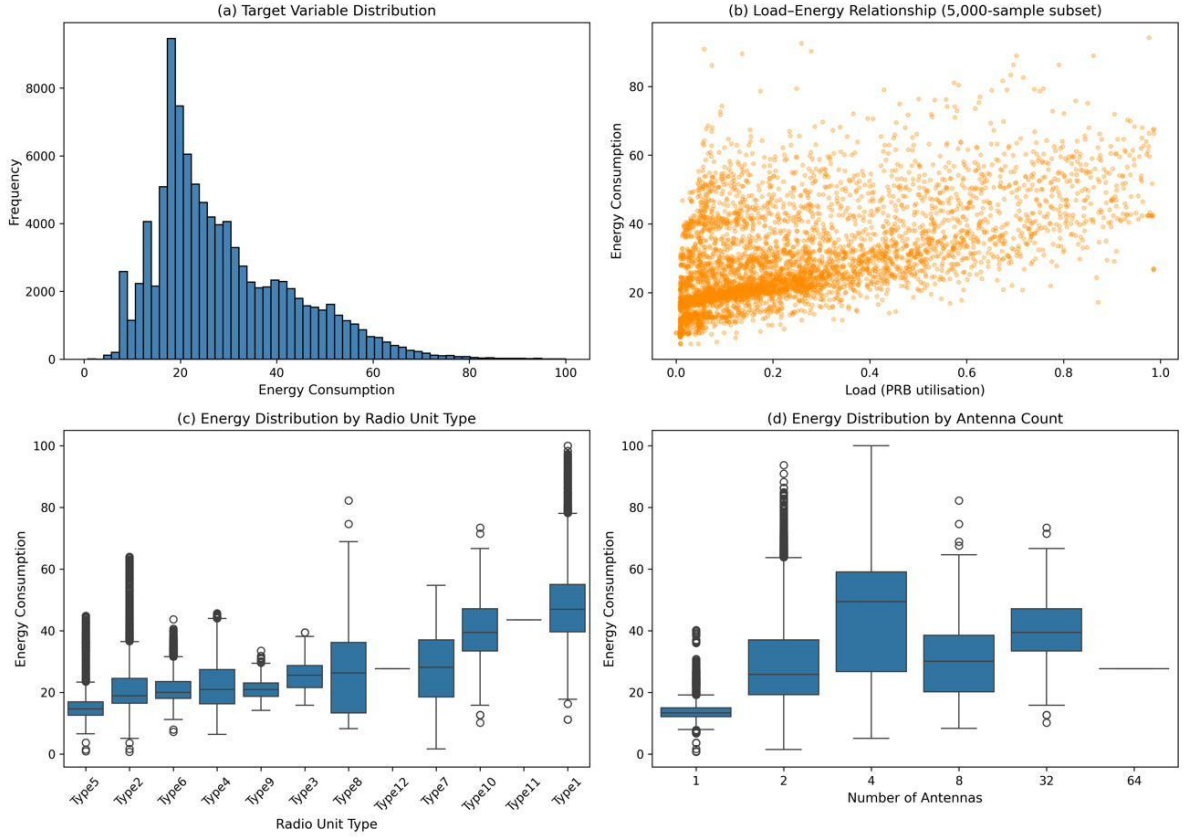

Figure 1. Overview of the training set: (a) distribution of the target variable, (b) relationship between traffic load and energy consumption (5,000-sample subset), (c) energy distribution by radio unit type, and (d) energy distribution by antenna count.

CC: BY-NC-ND

This is confirmed by Figures 1c and 1d. Energy consumption varies strongly with the radio unit type, ranging from a median near 14 for the lowest-consuming type to a median near 47, with a wide dispersion, for the highest. The relationship between the number of antennas and energy consumption is non-monotonic: four-antenna configurations exhibit a higher median than eight-antenna configurations, which indicates that antenna count interacts with other hardware attributes rather than determining energy consumption in isolation.

A diurnal pattern is present in the aggregate energy profile (Figure 2): mean consumption reaches its minimum during the early-morning hours (04:00–05:00) and its maximum in the late afternoon (around 18:00). However, the standard-deviation band is wide relative to the amplitude of the daily cycle, showing that variation across base stations dominates variation across the hours of the day. This observation motivates a feature-based modelling approach rather than a purely temporal one.

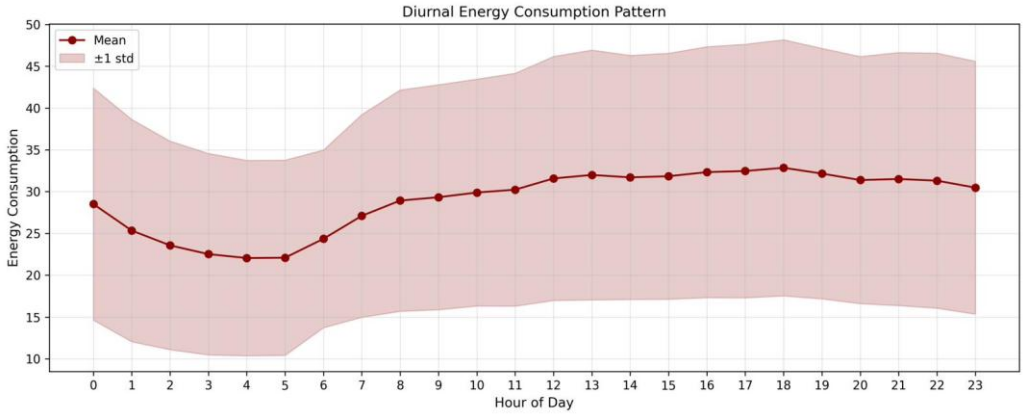

Figure 2. Hourly energy consumption pattern across the training set. The solid line denotes the hourly mean and the shaded band denotes one standard deviation.

A further property of the dataset, with direct implications for generalisation, concerns the difference in hardware composition between the labelled training stations and the stations reserved for evaluation. Figure 3 compares the antenna-count distribution of the two groups. The training stations are dominated by four-antenna configurations, whereas the evaluation-only stations are dominated by one- and two-antenna configurations and include eight-antenna configurations that are essentially absent from the training set. An analogous shift is observed for the categorical attributes: radio unit types Type11 and Type12, together with operating Mode1, are prevalent among the evaluation-only stations but nearly absent from the labelled training data. This distributional shift is acknowledged as a limitation of in-distribution evaluation and is revisited in the Discussion.

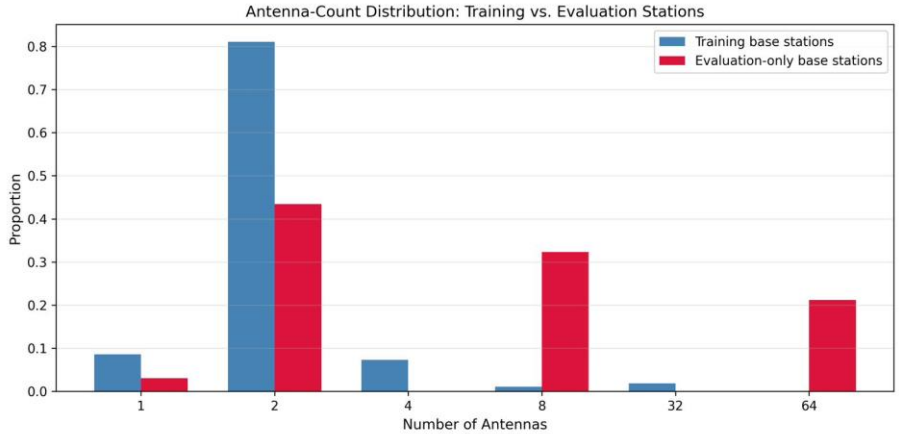

Figure 3. Antenna-count distribution of the labelled training base stations compared with the evaluation-only base stations, illustrating the hardware distribution shift between the two groups.

3.3. Feature Engineering

The raw predictors were augmented with a set of derived features to expose temporal structure and hardware–traffic interactions to the models. From the timestamp, the hour of day, the day of week, and the day index were extracted. To preserve the cyclic nature of the hour of day, a sine and cosine encoding was added, mapping hour h to $\sin(2\pi h/24)$ and $\cos(2\pi h/24)$ so that adjacent hours (for example 23:00 and 00:00) are represented as neighbouring points.

To capture the interaction between traffic and hardware capacity – motivated by the non-monotonic relationship observed in Figure 1d – three multiplicative interaction terms were constructed: load × antennas, load × transmission power, and load × bandwidth. Two summary features were derived from the six energy-saving mode signals: their sum (ES_total) and a binary indicator of whether any mode was active (ES_any_active).

An examination of the six energy-saving signals revealed that ESMode4 is identically zero across all 98,084 training records, whereas ESMode3 and ESMode5 are non-zero in only 168 records (0.17%, spanning 16 base stations) and 3 records (0.003%, spanning 2 base stations), respectively; ESMode1, ESMode2, and ESMode6 are active in approximately 4–5% of the records. As shown in the Discussion, the extreme sparsity of these signals causes the linear models to produce unstable extrapolations. These three signals were therefore removed from the predictor set, with their information retained in aggregate form through ES_total and ES_any_active.

The two categorical attributes, RUType and Mode, were encoded using one-hot encoding fitted on the union of categories present in the training and evaluation sets, yielding fourteen indicator columns. Combined with the eighteen numerical features, the final design matrix contains thirty-two features.

For clarity, the final design matrix comprises the following eighteen numerical features: the traffic load; four static hardware attributes (frequency, bandwidth, antenna count, and transmission power); the three retained energy-saving signals (ESMode1, ESMode2, ESMode6) together with their two aggregates (ES_total and ES_any_active); three interaction terms (load × antennas, load × transmission power, and load × bandwidth); and five temporal features (hour of day, day of week, day index, and the sine and cosine encodings of the hour). Combined with the fourteen one-hot indicators for RUType and Mode, this yields the thirty-two features used by all models.

3.4. Predictive Models

Seven predictors of increasing complexity were evaluated. A trivial baseline, which predicts the global mean of the training target for every observation, was included to establish the lower bound that any informative model must exceed. Two linear models were considered: ordinary linear regression and ridge regression with L2 regularisation ($\alpha = 1.0$). For the linear models, features were standardised to zero mean and unit variance, and predictions were clipped at zero to enforce the physical constraint that energy consumption is non-negative.

Two tree-based ensembles were evaluated: a random forest [8] (200 trees, minimum leaf size of two) and a gradient boosting model [9] implemented with LightGBM [11] (500 trees, learning rate 0.05, 63 leaves, early stopping after 30 rounds without validation improvement). Tree-based models do not require feature scaling and, by construction, cannot extrapolate beyond the range of the training target, which avoids the instability observed for the linear models. The trivial baseline, the linear models, and the random forest were implemented with scikit-learn [14], and LightGBM with its reference implementation.

Finally, a multilayer perceptron (MLP) was implemented in PyTorch [15]. The network consists of three hidden layers with 128, 64, and 32 units respectively, each followed by batch normalization [16], a rectified linear activation, and dropout [17] with rate 0.2, and a single linear output unit. It was trained with the Adam optimizer [18] (learning rate 10^{-3}, weight decay 10^{-5}) using the mean squared error loss, a batch size of 512, and early stopping after 15 epochs without validation improvement; the weights yielding the lowest validation loss were retained. The progressively narrowing three-hidden-

CC: BY-NC-ND

layer architecture (128-64-32) was chosen as a widely adopted configuration for tabular regression at this input dimensionality, balancing representational capacity against overfitting risk under batch normalization and dropout [13, 16, 17]; the fact that this network matches the accuracy of the tuned gradient-boosting model suggests that its capacity is adequate for the task.

Hyperparameter values for all models were set to commonly adopted configurations from the reference implementations and original sources of each method [11, 14, 18], rather than tuned via an exhaustive search procedure: the LightGBM configuration follows widely used practice for medium-sized tabular regression (500 trees with early stopping, learning rate 0.05, 63 leaves) [11], the random-forest and linear-model settings follow the scikit-learn reference implementation [14], and the MLP optimizer settings follow the defaults recommended for Adam [18]. The primary aim of this study is a controlled, transparent comparison of model families under a uniform evaluation protocol; a systematic hyperparameter optimization, for instance via Bayesian search, would likely yield further marginal gains for each model and is left for future work.

In the main experiments (Tables 3–6), the early-stopping criteria of LightGBM and the MLP were monitored on the held-out fold of each split, and for the MLP the weights with the lowest loss on that fold were retained. Because this choice uses the evaluation fold to select the number of boosting rounds or training epochs, its influence was quantified explicitly (Section 4.5, Table 7): monitoring early stopping instead on an inner validation split formed by 15% of the training stations changed R^2 by at most 0.001 and WMAPE by at most 0.07 percentage points. All additional analyses in Sections 4.5–4.7 use this inner-split procedure.

For the protocol-controlled comparison of Section 4.6, five existing techniques were re-implemented with the same thirty-two features: (i) an analytical load-dependent power model of EARTH type [23, 26], calibrated by least squares separately for each hardware configuration c as $E = a_c + b_c \cdot load + \Sigma_k g_{c,k} \cdot ESMode_k$ ($k \in \{1, 2, 6\}$), using the most specific configuration level with at least 100 training records from at least two stations (full configuration, then radio unit type × antenna count × bandwidth, then radio unit type, then all stations); (ii) XGBoost [10] (500 trees, learning rate 0.05, maximum depth 6); (iii) CatBoost [28] (up to 1,000 iterations, learning rate 0.05, depth 6); (iv) the hybrid of [21], in which XGBoost is fitted to the residuals of a ridge regression (the mixture-of-experts gate of [21] is tailored to the hidden challenge test set and was not reproduced); and (v) the weighted ensemble of [6], 0.3 × XGBoost + 0.3 × CatBoost + 0.4 × neural network, with the MLP of this study as the neural component. The boosting models use the same early-stopping rule (30 rounds) on the inner validation split.

3.5. Evaluation Protocol

Because the objective is to predict the energy consumption of base stations that may not have been observed during training, a conventional random train–test split would overestimate performance: records from the same base station would appear in both partitions, allowing a model to exploit station identity rather than learn generalisable relationships. To obtain a realistic estimate, a grouped five-fold cross-validation was adopted, in which the folds are partitioned by base station. In each fold, all records of a given station belong exclusively to either the training or the validation partition, so that every validation prediction is made for a station unseen during training. The partitioning yielded folds of approximately 185 stations each, with no station shared between training and validation in any fold.

Model performance was assessed using four complementary metrics. Let y_i denote the observed energy consumption, $\hat{y}_i$ the prediction, $\bar{y}$ the mean of the observed values, and n the number of observations. The mean absolute error (MAE) and the root mean squared error (RMSE) are defined as

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

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

CC: BY-NC-ND

The coefficient of determination (R^2) and the weighted mean absolute percentage error (WMAPE) are defined as

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \quad (3)$$

$$WMAPE = \frac{\sum_{i=1}^{n}|y_i - \hat{y}_i|}{\sum_{i=1}^{n}|y_i|} \times 100 \quad (4)$$

RMSE penalises large errors more heavily than MAE, and the comparison of the two is informative regarding the presence of occasional large deviations. WMAPE is reported primarily because it is the official evaluation metric of the ITU 5G Energy Consumption Modelling Challenge, which aids comparability with prior work on this dataset [3]; it is additionally robust to small target values, which inflate the conventional mean absolute percentage error. For each metric, the mean and standard deviation across the five folds are reported.

In addition to fold-level statistics, station-clustered bootstrap confidence intervals were computed on the out-of-fold predictions: the 923 base stations were resampled with replacement B = 10,000 times, and all metrics were recomputed on each resample. Clustering the resampling at the station level preserves the within-station dependence of hourly records and is consistent with the grouped cross-validation design.

Figure 4 summarizes how the results are validated. The first layer is the station-grouped cross-validation itself: because base stations are the units to which a deployed model must generalize, all records of a station are confined to one fold, in line with the recommendation to block cross-validation along the dependence structure of the data [31] and to avoid the leakage that arises when records of the same entity appear in both training and test data [22]. The second layer quantifies uncertainty: fold-level paired tests (t-test and Wilcoxon signed-rank test [33]) are complemented by the station-clustered bootstrap [32], which respects the dependence of hourly records within a station. The third layer consists of robustness checks that address specific threats to validity: (a) a record-level random split, which quantifies the optimism of a leaky protocol; (b) repetition of the grouped cross-validation with five different station partitions; (c) the placement of the early-stopping criterion; (d) the re-implementation of existing techniques under the identical protocol; (e) an out-of-time validation in which models trained on the first four days predict the remaining days [34]; (f) a traffic-shift stress test; and (g) a calendar-feature ablation. For these checks a single archived station-to-fold assignment is shared by all models, so that every comparison is paired.

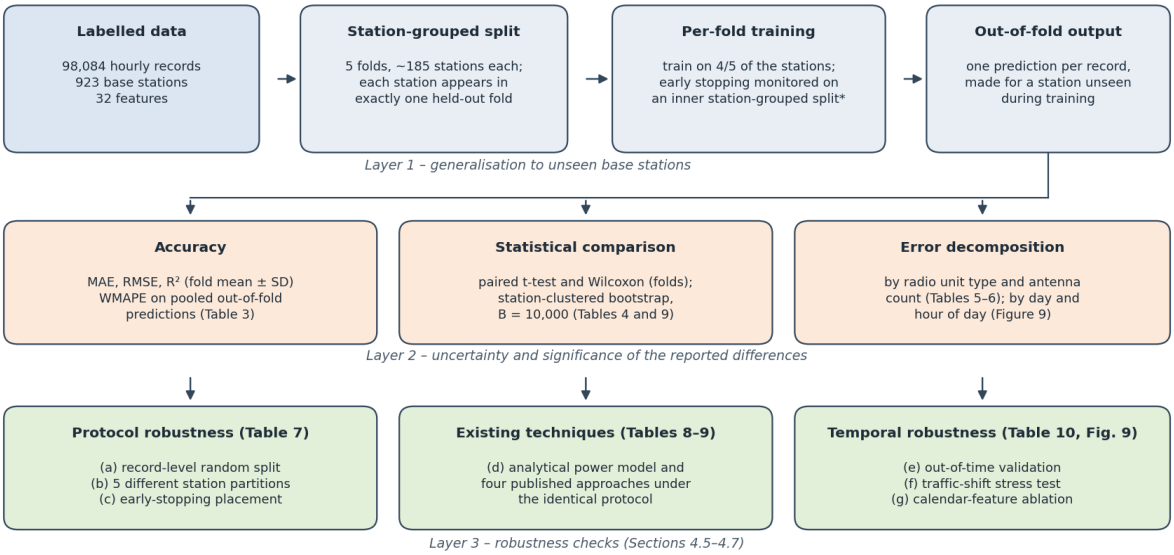

Figure 4. Validation framework of the study. Layer 1 estimates generalization to unseen base stations, Layer 2 quantifies the uncertainty of the reported differences, and Layer 3 comprises the robustness checks reported in Sections 4.5–4.7. *In the main experiments of Table 3, early stopping was monitored on the held-out fold; the effect of this choice is quantified in Table 7.

CC: BY-NC-ND

3.6. Ethical Approval

Ethical approval is not required for this study.

4. RESULTS

4.1. Comparative Performance

Table 3 reports the predictive performance of the seven models under the grouped five-fold cross-validation protocol. Each entry is the mean across the five folds, with the corresponding standard deviation. The trivial baseline, which predicts the global training mean, yields an R^2 of approximately zero and a WMAPE near 40%, confirming that it carries no predictive information and establishing the lower bound that informative models must exceed.

Table 3. Predictive performance of the seven models under grouped five-fold cross-validation (mean ± standard deviation across folds). Best values are shown in bold. Training time is the total wall-clock time across all five folds, measured on a single CPU instance (linear models, random forest, LightGBM) and on a Tesla T4 GPU (multilayer perceptron).

Model	MAE	RMSE	R^2	WMAPE (%)	Time (s)
Trivial (global mean)	11.495 ± 0.298	14.341 ± 0.551	−0.002 ± 0.002	39.66	< 1
Linear regression	3.648 ± 0.107	5.120 ± 0.049	0.872 ± 0.010	12.59	< 1
Ridge regression	3.648 ± 0.107	5.118 ± 0.048	0.872 ± 0.010	12.59	< 1
Linear regression ($\hat{y} \geq 0$)	3.646 ± 0.107	5.116 ± 0.048	0.872 ± 0.010	12.58	< 1
Random forest	3.132 ± 0.121	4.559 ± 0.238	0.898 ± 0.015	10.81	848
MLP (128-64-32)	2.996 ± 0.117	**4.268 ± 0.246**	**0.910** ± 0.014	10.34	281
LightGBM	**2.962** ± 0.142	4.271 ± 0.260	**0.910** ± 0.014	**10.22**	**14**

All non-trivial models substantially outperform this baseline. The two linear models achieve an R^2 of 0.872 and a WMAPE of approximately 12.6%. The zero-clipping treatment applied to the linear regression produces a negligible change in aggregate metrics relative to the unconstrained version, because the engineered feature set already removes the sparse signals responsible for unstable extrapolation (this behaviour is analysed in the Discussion). Ridge regression is indistinguishable from ordinary linear regression at this regularisation strength, indicating that overfitting is not a limiting factor for the linear models on this dataset.

The tree-based and neural models occupy the upper performance tier. Random forest improves the R^2 to 0.898 and reduces the WMAPE to 10.81%. LightGBM attains the best overall accuracy, with an R^2 of 0.910, an MAE of 2.962, and a WMAPE of 10.22%. The multilayer perceptron matches LightGBM in R^2 (0.910) and attains a marginally lower RMSE, although LightGBM retains a small advantage in MAE and WMAPE. Notably, LightGBM achieves this performance with a training time roughly twenty times shorter than that of the multilayer perceptron, a difference of practical significance for large-scale or frequently retrained deployments.

CC: BY-NC-ND

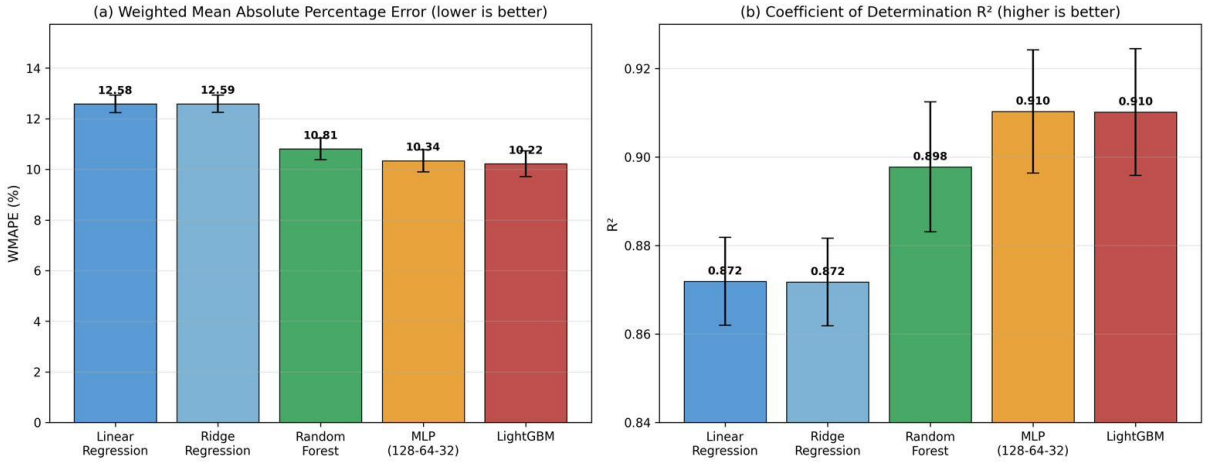

Figure 5. Comparison of the informative models: (a) weighted mean absolute percentage error and (b) coefficient of determination. Error bars denote one standard deviation across the five folds.

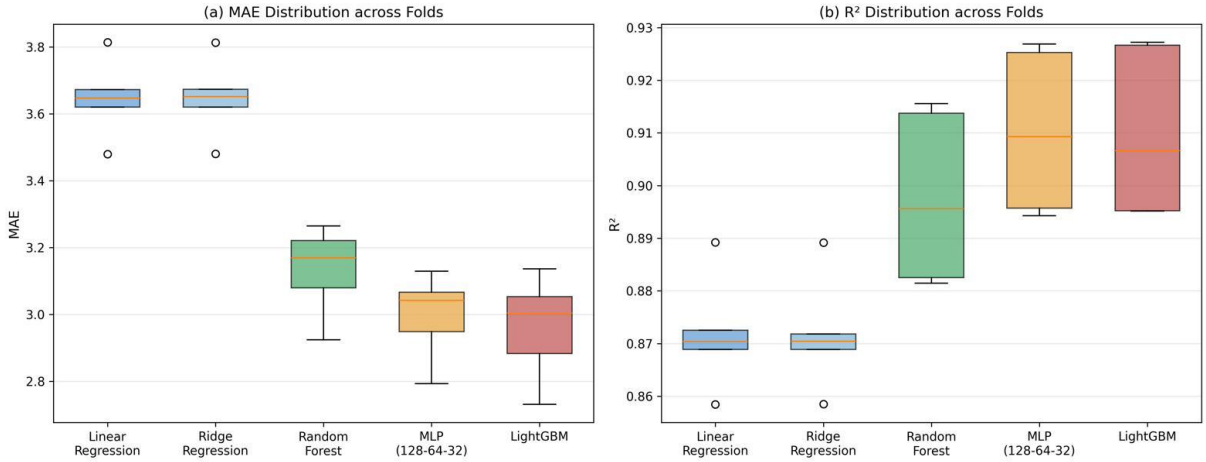

Figure 6. Fold-level distributions of (a) MAE and (b) R^2 for the informative models.

Figure 5 presents the same comparison visually for the informative models, with the trivial baseline omitted to preserve scale. Panel (a) shows the WMAPE and panel (b) the R^2, with error bars denoting the fold standard deviation. The clear separation between the linear models and the tree-based or neural models is evident, as is the near-equivalence of LightGBM and the multilayer perceptron. Figure 6 reports the corresponding fold-level distributions: the distributions of LightGBM and the multilayer perceptron largely overlap, consistent with their statistical indistinguishability, while both are clearly separated from the random forest and the linear models.

To assess whether the observed performance differences are statistically meaningful, paired comparisons were carried out using a paired t-test and a Wilcoxon signed-rank test at the fold level, a sample-level Wilcoxon test on the out-of-fold predictions, and station-clustered bootstrap confidence intervals. The results are summarized in Table 4. LightGBM significantly outperforms the linear baseline and the random forest in both R^2 and MAE, with the random-forest advantage being consistent across all five folds. The comparison with the multilayer perceptron, by contrast, yields no detectable difference in either R^2 or MAE: the two models are empirically indistinguishable on this dataset. At the sample level, the advantage of LightGBM over the linear baseline is statistically significant on the full set of 98,084 predictions ($p < 10^{-300}$), but the effect size remains small (Cohen's $d = -0.25$): LightGBM achieves a lower absolute error in 60.7% of the records, indicating that the aggregate accuracy gain, while meaningful in aggregate, is modest at the level of individual predictions. Station-clustered bootstrap intervals corroborate the fold-level picture: the LightGBM–MLP

CC: BY-NC-ND

differences are not distinguishable from zero (ΔMAE 95% CI [−0.068, +0.017]; ΔR^2 [−0.003, +0.002]), whereas the advantages over the random forest and the linear baseline are robust, with both intervals excluding zero.

Table 4. Paired statistical comparisons of LightGBM against the other informative models. Fold-level tests use the five cross-validation folds (n = 5); the sample-level comparison uses the out-of-fold predictions on all 98,084 records, with effect size reported as Cohen's d. Station-clustered bootstrap intervals are 95% percentile confidence intervals obtained by resampling the 923 base stations with replacement (B = 10,000) and recomputing the metric differences on the out-of-fold predictions; asterisks mark intervals excluding zero.

Comparison	Metric	Δ (mean)	paired t-test	Wilcoxon	Bootstrap 95% CI
Fold level (n = 5)					
LightGBM vs. Linear regression	R^2	+0.038	p = 0.002	p = 0.063 [a]	[+0.028, +0.047] *
LightGBM vs. Linear regression	MAE	−0.685	$p < 10^{-3}$	p = 0.063 [a]	[−0.779, −0.570] *
LightGBM vs. Random forest	R^2	+0.012	$p < 10^{-4}$	p = 0.063 [a]	[+0.010, +0.014] *
LightGBM vs. Random forest	MAE	−0.170	$p < 10^{-3}$	p = 0.063 [a]	[−0.196, −0.127] *
LightGBM vs. MLP	R^2	−0.0004	p = 0.76	p = 1.00	[−0.003, +0.002]
LightGBM vs. MLP	MAE	−0.026	p = 0.28	p = 0.31	[−0.068, +0.017]
Sample level (n = 98,084, out-of-fold)					
LightGBM vs. Linear regression	MAE	−0.672	—	$p < 10^{-300}$	d = −0.25 (small)

[a] For n = 5 paired observations, the two-sided Wilcoxon signed-rank test has a minimum attainable p-value of 0.0625; comparisons reaching this value indicate that all five folds favored the same model.

4.2. Feature Importance

To identify the drivers of the predictions, permutation importance was computed for the best-performing model (LightGBM) on a held-out fold using the implementation in scikit-learn [14], measuring the increase in MAE when each feature is randomly permuted. Figure 7 reports the twelve most important features. Two features dominate: the indicator for radio unit type Type1 and the engineered load × antennas interaction term. Permuting the former increases the MAE by more than 6 units – over twice the model's baseline MAE – while permuting the latter increases it by approximately 4 units. The remaining features, including transmission power, antenna count, frequency, and the hour of day, contribute comparatively modest individual effects.

Two observations follow. First, the prominence of the engineered load × antennas term confirms the value of the interaction features: the model relies more heavily on this composite quantity than on either constituent variable in isolation, consistent with the non-monotonic antenna–energy relationship observed in Figure 1d. Second, the concentration of importance in hardware-related features indicates that energy consumption is governed primarily by the static hardware configuration, with traffic load acting through its interaction with hardware capacity rather than as an independent driver.

CC: BY-NC-ND

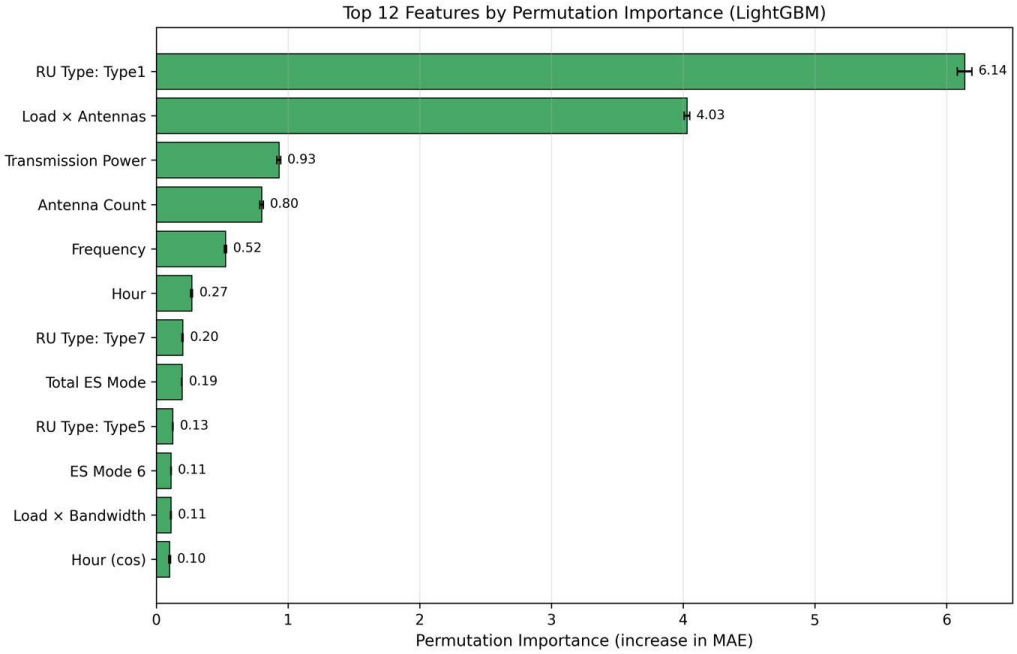

Figure 7. Twelve most important features for the LightGBM model, ranked by permutation importance (increase in MAE when the feature is permuted). Error bars denote the standard deviation over ten permutation repeats.

4.3. Prediction Quality

Figure 8 compares the out-of-fold predictions of the linear regression and LightGBM models against the observed energy consumption. Each point corresponds to a prediction made for a base station that was not seen during the training of the respective fold. The linear model (panel a) exhibits a systematic bias in the low-energy region, where predictions tend to exceed the observed values, and an increasing dispersion in the high-energy region. The LightGBM model (panel b) produces predictions that adhere more closely to the identity line across the full range, with markedly reduced dispersion in the low-to-moderate energy region. This visual comparison is consistent with the aggregate metrics in Table 3.

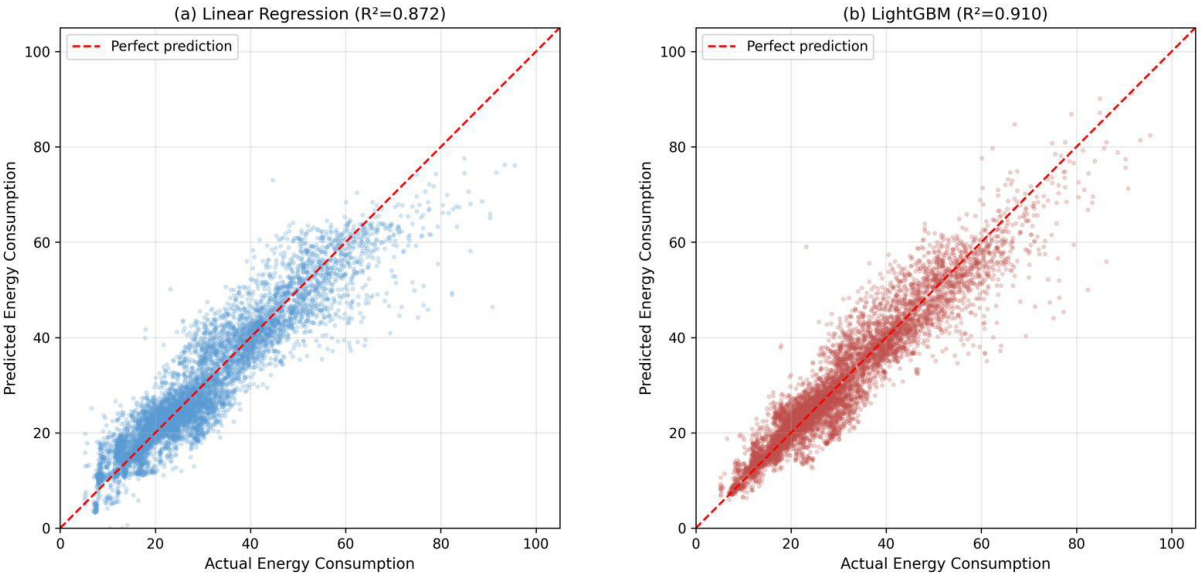

Figure 8. Out-of-fold predicted versus actual energy consumption for (a) linear regression and (b) LightGBM. The dashed line denotes perfect prediction. An 8,000-point random subset is shown for clarity.

CC: BY-NC-ND

4.4. Per-Segment Error Analysis

To examine whether prediction quality is uniform across the heterogeneous hardware population, the out-of-fold predictions were further disaggregated by radio unit type and by antenna count. Table 5 reports per-segment metrics for LightGBM and the linear baseline, together with the segment's share of the training data and its mean energy level.

Three observations emerge. First, prediction accuracy does not follow the share of training examples in a simple manner. The two most prevalent radio unit types, Type1 and Type4 (together accounting for over half of the training records), occupy opposite ends of the accuracy spectrum: Type4 achieves the lowest MAE among all segments (MAE = 1.61, R^2 = 0.92), whereas Type1 yields the highest MAE (MAE = 5.02, R^2 = 0.69). Conversely, sparsely represented segments such as Type8 and Type10 (each below 0.5% of the data) attain high R^2 values (0.95 and 0.96 respectively). Segment-level accuracy therefore reflects the internal consistency of each hardware class rather than the volume of training examples it contributes. The elevated error on Type1 is consistent with its mean energy level being roughly twice that of any other type (47.88 versus a median of ∼25 across other types) and with the wide energy dispersion observed for this type in Figure 1c, suggesting that Type1 encompasses a heterogeneous mixture of operating regimes that the model does not fully resolve.

Second, the advantage of LightGBM over the linear baseline varies markedly across segments. In the most prevalent segments, the MAE difference is modest (−0.40 for Type1, −0.25 for Type6). In sparsely represented segments, however, LightGBM's advantage becomes pronounced: −3.71 MAE units for Type8, −4.74 for Type10, and −2.09 for Type5. This pattern is consistent with the inability of linear models to extrapolate to under-represented regions of the input space: the tree-based predictor's bounded outputs and its ability to assign separate leaf estimates to rare configurations protect it where the linear model fails. The same trend is visible in Table 6, where the LightGBM advantage grows from −0.42 MAE for the dominant two-antenna configuration to −3.04 for eight-antenna and −4.74 for thirty-two-antenna configurations.

Third, the R^2 metric should be interpreted with care at the segment level. The one-antenna group exhibits a low R^2 of 0.42 despite a small MAE of 1.79; this reflects the narrow energy range within the segment (mean 14.08, low variance), which inflates the proportional penalty in the R^2 formula. MAE and WMAPE are more informative for within-segment comparisons under heterogeneous variance.

Table 5. Per-segment error analysis by radio unit type. MAE values for LightGBM (LGB) and the linear baseline are reported on the out-of-fold predictions. The MAE difference column indicates by how much LightGBM reduces the linear baseline's error in each segment.

RU Type	*n*	Share (%)	$\bar{y}$	MAE (LGB)	R^2 (LGB)	WMAPE (LGB, %)	ΔMAE (LGB−Lin)
Type4	25,677	26.18	21.71	1.61	0.92	7.42	−0.64
Type1	24,195	24.67	47.88	5.02	0.69	10.48	−0.40
Type6	22,083	22.51	21.10	2.10	0.63	9.96	−0.25
Type7	12,641	12.89	28.35	3.87	0.82	13.64	−0.52
Type3	5,462	5.57	25.23	3.23	0.26	12.80	−1.88
Type2	3,861	3.94	21.27	2.56	0.88	12.01	−2.08
Type5	2,992	3.05	16.50	1.31	0.94	7.92	−2.09
Type9	518	0.53	21.28	1.15	0.80	5.41	−0.12
Type10	403	0.41	39.08	1.75	0.96	4.49	−4.74
Type8	250	0.25	27.01	2.13	0.95	7.87	−3.71

CC: BY-NC-ND

Table 6. Per-segment error analysis by antenna count.

Antennas	*n*	Share (%)	$\bar{y}$	MAE (LGB)	R^2 (LGB)	WMAPE (LGB, %)	ΔMAE (LGB−Lin)
1	8,412	8.58	14.08	1.79	0.42	12.70	−1.55
2	82,246	83.85	29.03	3.03	0.89	10.42	−0.42
4	6,811	6.94	46.27	3.88	0.93	8.39	−2.34
8	211	0.22	29.88	1.75	0.96	5.85	−3.04
32	403	0.41	39.08	1.75	0.96	4.49	−4.74

4.5. Robustness of the Validation

The robustness checks (a)–(c) of Figure 4 are summarized in Table 7. For these checks, all models were re-run in a single pass with one archived station-to-fold assignment and with early stopping monitored on an inner validation split of the training stations (Section 3.4). In this reference configuration LightGBM attains $R^2 = 0.907$ and WMAPE = 10.51%, which differ from the values in Table 3 (0.910 and 10.22%) by less than the fold-to-fold standard deviation and therefore reflect the random assignment of stations to folds rather than a change in modelling.

(a) A record-level random split, in which hourly records of the same base station appear in both training and test folds, raises the apparent accuracy of LightGBM to $R^2 = 0.918$ and WMAPE = 9.67%, and XGBoost and CatBoost reach a similar level. The resulting optimism of 0.74–0.84 percentage points is moderate because the feature set contains no station identifier; Section 4.6 shows that it becomes much larger when station-specific information is available, which explains why the errors reported in the literature for this dataset are not comparable with unseen-station estimates.

(b) Across five different random station partitions (25 folds in total), LightGBM attains $R^2 = 0.905 \pm 0.001$ and WMAPE = 10.51 ± 0.06% (mean ± standard deviation over the partitions). The partition-to-partition variation is small compared with the differences between model families: in every partition the four boosting-based techniques lie within 0.24 percentage points of each other in WMAPE, and all of them remain at least 1.06 points more accurate than the analytical model, which in turn remains more accurate than linear regression. The MLP is less accurate than LightGBM in all five partitions (ΔWMAPE between +0.20 and +0.45 points).

(c) Monitoring early stopping on the held-out fold, as in Table 3, changes the results only marginally: R^2 increases by 0.0008 for LightGBM and by 0.0009 for the MLP, and WMAPE changes by −0.07 and +0.02 percentage points, respectively. Using the held-out fold to select the number of boosting rounds or epochs therefore does not affect the conclusions drawn from Table 3, although the inner-split procedure is the methodologically cleaner choice and is used for all additional analyses.

CC: BY-NC-ND

Table 7. Robustness of the validation protocol. MAE and R^2 are means ± standard deviations across folds; WMAPE is computed on the pooled out-of-fold predictions (for the repeated partitions: mean ± standard deviation across the five partitions). Unless stated otherwise, early stopping is monitored on an inner validation split of the training stations.

Validation variant / model	MAE	R^2	WMAPE (%)
Reference: station-grouped five-fold CV, archived partition			
LightGBM	3.048 ± 0.147	0.907 ± 0.010	10.51
MLP (128-64-32)	3.099 ± 0.125	0.904 ± 0.008	10.69
(a) Record-level random five-fold CV (records of a station shared between folds)			
Linear regression	3.619 ± 0.015	0.876 ± 0.007	12.48
LightGBM	2.803 ± 0.013	0.918 ± 0.002	9.67
XGBoost	2.790 ± 0.010	0.918 ± 0.002	9.62
CatBoost	2.798 ± 0.013	0.918 ± 0.002	9.65
(b) Station-grouped CV repeated over five station partitions (25 folds)			
Linear regression	3.689 ± 0.182	0.870 ± 0.020	12.73 ± 0.07
Analytical load model (EARTH type)	3.418 ± 0.185	0.881 ± 0.018	11.79 ± 0.04
LightGBM	3.046 ± 0.159	0.905 ± 0.016	10.51 ± 0.06
XGBoost	3.026 ± 0.168	0.906 ± 0.016	10.44 ± 0.08
CatBoost	3.022 ± 0.163	0.907 ± 0.016	10.43 ± 0.05
Ridge + XGBoost hybrid	3.051 ± 0.157	0.901 ± 0.019	10.53 ± 0.08
MLP (128-64-32)	3.153 ± 0.180	0.899 ± 0.016	10.88 ± 0.09
(c) Early stopping monitored on the held-out fold (protocol of Table 3)			
LightGBM	3.028 ± 0.115	0.908 ± 0.010	10.45
MLP (128-64-32)	3.105 ± 0.128	0.905 ± 0.008	10.71

4.6. Comparison with Existing Techniques

Table 8 summarizes the results reported for the ITU dataset in the literature. The reported errors span a wide range, from a MAPE of 3.56% [6] to a WMAPE of 13.91% [4], but they were obtained under markedly different protocols. The lowest errors arise when the test records belong to stations that are also present in the training data, either because the cross-validation is performed at record level [6] or because the test set contains known stations whose identifier is used as an input [4]. For stations not encountered during training, Chen et al. report a WMAPE of 6.69% for their best model and 8.16% without the station identifier, and their re-implementation of the model of [2] yields 13.91% [4]; on the hidden challenge test set, which emphasizes new products and configurations, the hybrid of [21] reaches 6.99%, while XGBoost and ridge regression alone reach 9.68% and 9.55%. The unseen-station WMAPE of 10.22% obtained here (Table 3) therefore lies within the range of 6.7–13.9% reported for settings that involve unseen stations.

CC: BY-NC-ND

Table 8. Results reported in the literature for the ITU 5G energy dataset. The error measure of [4] is $\Sigma|y - \hat{y}|/\Sigma|y|$ (their Eq. 4), i.e. the WMAPE used here, although it is labelled MAPE there; [6] reports the conventional MAPE; [21] reports the weighted WMAPE of the challenge leaderboard.

Study	Technique	Evaluation protocol	Test stations	Reported error (%)
Chen et al. [4]	MLP with station-identifier embedding, attention and masked training	Official challenge split	Seen / unseen	4.00 / 6.69
Chen et al. [4]	Same MLP without station identifier	Official challenge split	Seen / unseen	9.22 / 8.16
Chen et al. [4]	Re-implementation of the ML power model of [2]	Official challenge split	Seen / unseen	11.39 / 13.91
Makroz et al. [6]	Weighted ensemble of XGBoost, CatBoost and ANN (0.3/0.3/0.4)	Record-level 5-fold CV	Seen	3.56 (MAPE)
Zimmermann [21]	Ridge + XGBoost hybrid with mixture-of-experts gate	Hidden challenge test set	Unseen (new products and configurations)	6.99
Zimmermann [21]	XGBoost alone / ridge regression alone	Hidden challenge test set	Unseen (new products and configurations)	9.68 / 9.55
This study (Table 3)	LightGBM	Station-grouped 5-fold CV	Unseen	10.22
This study (Section 4.7)	LightGBM with station fingerprint	Out-of-time, stations seen in the training period	Seen	4.96

To remove the influence of the protocol, the existing techniques were re-implemented and evaluated under the identical station-grouped protocol (Table 9). The analytical load-dependent model, calibrated separately for each hardware configuration, improves on linear regression (WMAPE 11.76% versus 12.66%), confirming that a configuration-specific linear load model captures a substantial part of the variation [23, 26], but it remains clearly less accurate than the tree-based learners (ΔWMAPE = +1.25 percentage points relative to LightGBM, 95% CI [+1.06, +1.45]). The gradient-boosting implementations are practically equivalent: XGBoost and CatBoost attain WMAPE values of 10.42% and 10.39%, against 10.51% for LightGBM, and the ridge–XGBoost hybrid of [21] performs on par (10.45%). The ensemble of [6] is the most accurate technique (WMAPE 10.36%), but its advantage over LightGBM is small (ΔWMAPE = −0.15 percentage points; 95% CI [−0.24, −0.07]) and not significant at the fold level (paired t-test on MAE, p = 0.13), whereas its training time is about 30 times larger.

Under this partition the MLP is slightly less accurate than LightGBM (ΔR^2 = −0.003; 95% CI [−0.006, −0.001]; fold-level p = 0.11), a difference that was stable across three random seeds (R^2 between 0.9036 and 0.9042) but did not appear in Table 3. Across the five additional station partitions of Table 7, the MLP was less accurate than LightGBM in every partition (ΔR^2 between −0.008 and −0.003), so the near-equivalence observed in Table 3 does not hold across station partitions. This strengthens rather than changes the practical conclusion, given the twentyfold difference in training cost.

The station-fingerprint principle of [4] cannot be applied to unseen stations, because no station-specific information exists for them. It can, however, be examined in the out-of-time setting of Section 4.7, in which the test stations are observed during the training period: adding the mean energy and mean load of each station over the training days to the LightGBM features reduces the WMAPE on the

CC: BY-NC-ND

subsequent days from 9.77% to 4.96% (R^2 from 0.917 to 0.976). This value matches the errors reported for known stations in [4] (4.00%) and for record-level cross-validation in [6] (3.56%), and it reconciles the literature: the low errors reported previously reflect the availability of station-specific history, whereas the level of about 10% obtained here characterizes stations without measurement history, the setting that matters when new deployments and configurations are planned.

Table 9. Existing techniques re-implemented under the identical station-grouped protocol (archived partition, early stopping on an inner validation split). ΔWMAPE is the difference to LightGBM with its 95% station-clustered bootstrap interval (B = 10,000); training time is the total over five folds on a single CPU core.

Technique	Source	MAE	R^2	WMAPE (%)	ΔWMAPE vs. LightGBM [95% CI]	Time (s)
Linear regression	This study	3.669 ± 0.132	0.872 ± 0.003	12.66	+2.14 [+1.79, +2.52]	< 1
Analytical load model (EARTH type)	[23, 26]	3.410 ± 0.133	0.883 ± 0.010	11.76	+1.25 [+1.06, +1.45]	40[b]
XGBoost	[10]	3.019 ± 0.115	0.908 ± 0.010	10.42	−0.10 [−0.18, −0.02]	9
CatBoost	[28]	3.013 ± 0.124	0.909 ± 0.010	10.39	−0.12 [−0.20, −0.04]	27
Ridge + XGBoost hybrid	[21]	3.029 ± 0.111	0.904 ± 0.005	10.45	−0.06 [−0.17, +0.04]	9
Weighted ensemble (XGBoost, CatBoost, ANN)	[6]	3.004 ± 0.122	0.909 ± 0.010	10.36	−0.15 [−0.24, −0.07]	176
MLP (128-64-32)	This study	3.099 ± 0.125	0.904 ± 0.008	10.69	+0.18 [+0.06, +0.29]	140
LightGBM	This study	3.048 ± 0.147	0.907 ± 0.010	10.51	–	6

[b] Unoptimized Python implementation of the per-configuration fits; not indicative of the computational cost of the model.

4.7. Temporal Robustness and Traffic Shift

Table 10 examines whether the accuracy obtained within the observed week can be expected to carry over to other periods. Removing the day index and day of week, which are specific to the observed week, leaves the accuracy of LightGBM virtually unchanged (WMAPE 10.57% versus 10.51%), whereas removing all calendar features, including the hour of day, increases it to 11.41%. The model therefore does not depend on date-specific information, while the hour of day carries information beyond the traffic load itself, most likely through the time-of-day pattern of energy-saving activation. In the out-of-time validation, models trained on the first four days (1–4 January) predicted the remaining days (5–8 January) for stations not used in training, with the day index and day of week omitted. Compared with predicting the same records from a model trained on all days, temporal extrapolation costs between 0.07 and 0.13 percentage points of WMAPE across all models.

Figures 9a and 9b break down the out-of-fold error of LightGBM by calendar day and by hour. The daily WMAPE varies only between 10.2% and 11.1%, and the error on the weekend days (10.46%) is indistinguishable from that on weekdays (10.58%). By contrast, the hourly WMAPE is highest in the

CC: BY-NC-ND

low-traffic night hours (up to 13.7% at 01:00) and lowest in the afternoon (9.4%), and it is negatively correlated with the mean load (r = −0.73). Relative errors are therefore largest when consumption is low and energy-saving modes are most frequently active (in 21–24% of the records between 02:00 and 07:00, against 4–5% during the day).

The traffic-shift stress test (Figure 9c) emulates a period with heavier traffic than the training period: within each fold, the models were trained only on records whose load did not exceed the 80th percentile of the training load and were evaluated on the held-out stations' records above that threshold (19,643 records). Trained on all hours, the boosting models are the most accurate on these records (WMAPE 8.59–8.77%). Trained on low-load hours only, they degrade most strongly (WMAPE 12.07–13.34%) and systematically underestimate consumption (mean bias −4.2 to −2.8), because, as noted in Section 3.4, tree-based models cannot extrapolate beyond the range of the training target. Linear regression and the analytical model degrade less (to 10.60% and 11.41%), and the ridge–XGBoost hybrid of [21], whose linear component extrapolates the load trend, is the most robust technique (10.22%). The MLP takes an intermediate position (10.76%, mean bias −2.3): its piecewise-linear activations allow some extrapolation, but less than the explicit linear component of the hybrid. The ranking of the model families obtained within the observed traffic range is thus not preserved under a shift of the load distribution.

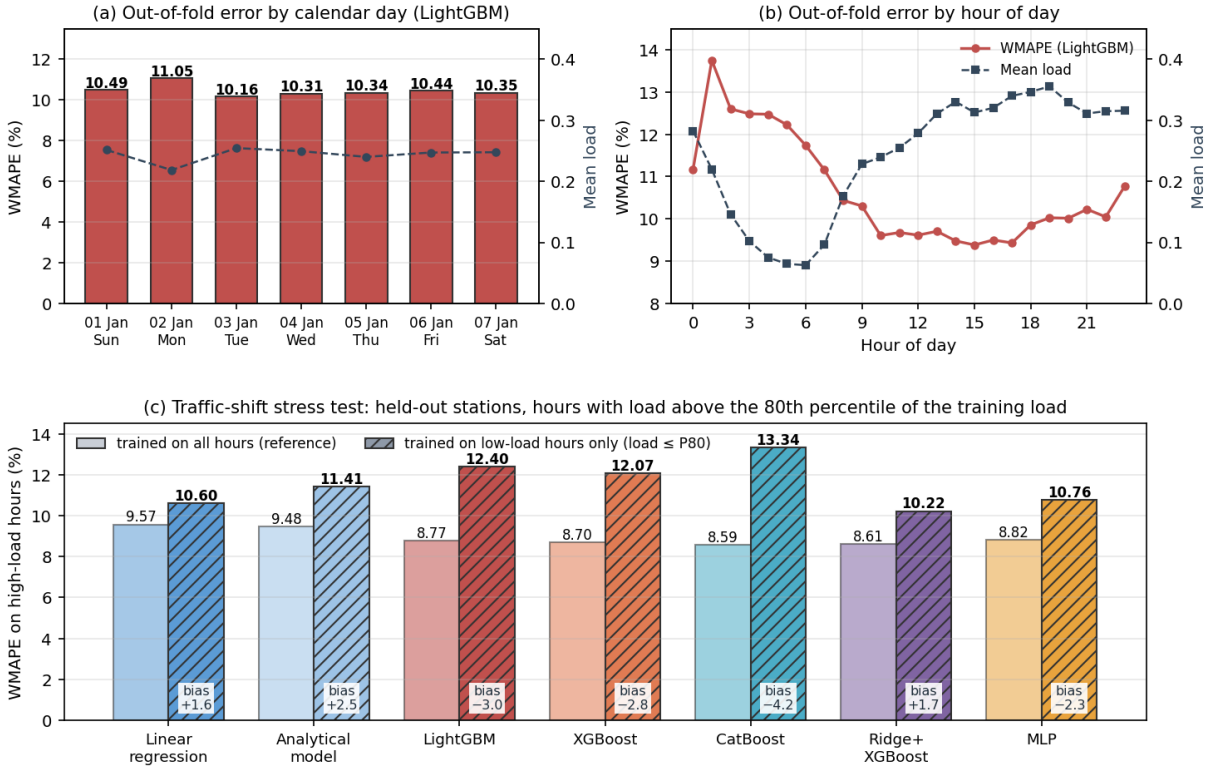

Figure 9. Temporal robustness of the predictions. (a) Out-of-fold WMAPE of LightGBM by calendar day (bars) and mean traffic load (dashed line); 8 January, which contains only 44 records at 00:00, is omitted. (b) Out-of-fold WMAPE by hour of day and mean load. (c) Traffic-shift stress test: WMAPE on the held-out stations' hours whose load exceeds the 80th percentile of the training load, for models trained on all hours (light bars) and on low-load hours only (hatched bars); "bias" is the mean signed error (prediction − observation) of the latter.

Table 10. Temporal robustness. Calendar ablation: station-grouped CV on all records (LightGBM). Out-of-time: models trained on 1–4 January and evaluated on 5–8 January, either for unseen stations (reference: the same records predicted by a model trained on all days) or for stations seen in the training period (reference: LightGBM without fingerprint).

Scenario / model	Reference WMAPE (%)	Scenario WMAPE (%)	Δ (percentage points)
Calendar-feature ablation (LightGBM, 98,084 records)			
All 32 features (reference)	–	10.51	–
Without day index and day of week	10.51	10.57	+0.05
Without any calendar feature	10.51	11.41	+0.90
Out-of-time validation, unseen stations (31,340 test records)			
Linear regression	12.34	12.45	+0.12
Analytical load model (EARTH type)	11.56	11.63	+0.07
LightGBM	10.47	10.57	+0.10
XGBoost	10.35	10.48	+0.13
CatBoost	10.37	10.49	+0.13
Out-of-time validation, stations seen in the training period (31,340 test records)			
LightGBM	–	9.77	–
LightGBM with station fingerprint	9.77	4.96	−4.81

5. DISCUSSION

The results in Section 4 support several observations regarding the prediction of 5G base-station energy consumption, the relative merits of the model families considered, and the limitations of the present evaluation.

5.1. Hardware Configuration as the Dominant Predictor

The feature-importance analysis (Figure 7) and the load–energy relationship (Figure 1b) together indicate that energy consumption is governed primarily by the static hardware configuration of a base station rather than by its instantaneous traffic load. The radio-unit-type indicator and the engineered load × antennas interaction term jointly account for most of the model's predictive capacity, while traffic load in isolation is a comparatively weak predictor. This finding is consistent with the wide dispersion observed in the load–energy scatter, where a given load level corresponds to a broad range of energy values, and with the non-monotonic relationship between antenna count and energy consumption (Figure 1d). It also aligns with prior work emphasising the importance of per-station characteristics [4]: the dominant role of hardware-related features explains why models that omit such information suffer from the one-to-many mapping between inputs and outputs. The per-segment analysis in Section 4.4 further refines this picture: while the radio-unit-type indicator dominates aggregate importance, segment-level accuracy is largely independent of the segment's training share, and the most pronounced gains of LightGBM over the linear baseline occur in the under-represented hardware categories where linear extrapolation is least reliable. From a practical standpoint, this implies that energy-saving strategies and predictive models alike should treat hardware configuration as a first-class input, and that the value of the engineered interaction term demonstrates the benefit of encoding hardware–traffic coupling explicitly.

5.2. Accuracy, Stability, and Computational Cost

Gradient boosting (LightGBM) and the multilayer perceptron achieved the highest accuracy, with no detectable difference in either R^2 (paired t-test $p = 0.76$) or MAE ($p = 0.28$; station-clustered bootstrap 95% CI for the MAE difference [−0.068, +0.017], containing zero; Table 4). LightGBM

CC: BY-NC-ND

retained a small advantage in MAE and WMAPE, while the MLP attained a marginally lower RMSE. This near-equivalence is itself noteworthy: the broader tabular-data literature reports that tree-based models frequently outperform neural networks on medium-sized datasets [7], whereas here a conventional multilayer perceptron, supported by appropriate feature engineering and normalization, matched the gradient boosting model in R^2. This equivalence is, however, sensitive to the station partition: in each of the further partitions used for the additional analyses, the MLP was slightly less accurate than LightGBM (Sections 4.5–4.6). A decisive difference, however, lies in computational cost: LightGBM completed five-fold training in 14 seconds on a CPU, compared with 281 seconds for the multilayer perceptron on a GPU and 848 seconds for the random forest on a CPU (Table 3) – a roughly twentyfold and sixtyfold gap respectively. For operational settings in which models are retrained frequently or applied across very large base-station populations, this cost asymmetry is of direct practical relevance and favours gradient boosting as the default choice, with the neural network reserved for scenarios where its representational flexibility can be leveraged further. The protocol-controlled comparison with existing techniques (Table 9) reinforces this conclusion: the boosting implementations and the published ensemble of [6] differ by at most 0.15 percentage points in WMAPE, a margin that is small compared with the fold-to-fold variability and with the differences in training cost, consistent with the observation that combining neural networks with boosted trees yields small but consistent gains on tabular data [29].

5.3. Instability of Linear Models under Sparse Signals

A methodological finding concerns the behavior of the linear models. The energy-saving mode signals are highly sparse: one mode is identically zero throughout the dataset, and two others are active in only a handful of records. When such sparse signals are standardised, their rare non-zero occurrences are mapped to extreme standardised values, and the linear models respond with large coefficients that, on rare activations, produce physically impossible predictions, including substantially negative energy values. This instability was the primary source of elevated error in the linear models prior to mitigation. Two simple measures resolved it: aggregating the sparse mode signals into summary features and removing the individual sparse columns, and clipping predictions at zero to enforce the non-negativity of energy. These measures stabilized the linear baseline without recourse to more complex modelling. Tree-based models were not subject to this issue, as their predictions are bounded by the range of the training target by construction. This observation is a practical caution for the use of linear models on operational network data, where sparsely activated control signals are common.

5.4. Temporal Validity and Seasonal Considerations

Whether the reported energy analysis remains valid throughout the year cannot be answered directly with the present data, which cover a single week whose season is not unambiguously documented (Section 3.1). The analyses of Section 4.7 nevertheless allow the question to be separated into parts that can and cannot be assessed. The models learn a mapping from hardware configuration, traffic load and energy-saving activity to energy consumption; they are not time-series forecasts. The static hardware attributes, which dominate the feature importance (Section 4.2), do not change with the season, the models do not rely on date-specific information (Table 10), and within the observed period the accuracy is stable across days, including the weekend, while extrapolating from four days to the following days costs at most 0.13 percentage points.

Three mechanisms may nevertheless change the energy analysis over the year. First, mobile traffic shows strong daily and weekly regularities but also longer-term trends and deviations caused by events and holidays [35, 36], so the load distribution of another season may differ from that of the training period. Seasonal traffic enters the models as an input, yet the stress test of Section 4.7 shows that tree ensembles underestimate consumption when the load exceeds the training range (WMAPE rising from 8.77% to 12.40% for LightGBM), a form of covariate shift [37]. Second, ambient temperature affects the energy use of a site: measurements show that air conditioning of equipment shelters can account for a large share of site consumption [27]. The dataset contains no temperature information and its normalized energy values do not document whether cooling is included, so

CC: BY-NC-ND

temperature-driven seasonal variation is not represented in the present models. Third, operators may adapt the activation policies of energy-saving modes over the year; because these activations are model inputs, such changes are captured as long as the corresponding modes were observed during training.

Accordingly, the hardware-related findings of this study – the dominance of the static configuration, the relative standing of the model families within the observed load range and the instability of linear models under sparse signals – are expected to hold across seasons, whereas the absolute error level may change in periods with traffic outside the training range or with strong thermal effects. In practice this calls for periodic recalibration, for which the low training cost of LightGBM (seconds per model) is a decisive advantage, for monitoring of input drift, and for hybrid models with a linear component when traffic is expected to exceed the training range [21]; ambient temperature should be added as an input when it is available. A definitive answer requires multi-season measurements, which are not publicly available for this dataset.

5.5. Limitations

The principal limitation of this study concerns distributional shift. As shown in Figure 3, the base stations reserved for the challenge evaluation differ markedly from the labelled training stations in their hardware composition: antenna counts, certain radio-unit types, and one operating mode that are prevalent among the evaluation stations are nearly absent from the training data. Because the energy labels for the evaluation partition are not disclosed, the performance of the models under this hardware distribution shift could not be measured directly, and the reported metrics therefore reflect in-distribution generalization to unseen base stations rather than generalization under distribution shift. This is an important distinction: the challenge was explicitly designed to reward cross-equipment and cross-configuration generalization [3], and strong in-distribution performance does not guarantee robustness when the hardware distribution changes.

A synthetic out-of-distribution evaluation – in which selected hardware categories are held out from training and used exclusively for testing – was considered as an alternative means of quantifying this risk on the labelled portion of the data. However, the rare radio unit types (Type8 through Type12) jointly comprise only 1.2% of the labelled records, and Type11 and Type12 contain only a single labelled record each. A synthetic split based on these categories would therefore yield a test set that is both too small and too unbalanced to support statistically reliable conclusions, and could not approximate the markedly different hardware composition observed in the official evaluation partition. We therefore chose to characterize the distribution shift descriptively (Figure 3) and to defer its quantitative evaluation to future work using independently collected operational data. Quantifying and improving robustness under such shift is left as future work, for instance through domain-adaptation techniques, the explicit modelling of per-station characteristics [4], or validation on independently collected operational datasets. A further limitation is that the dataset spans a single week whose season is uncertain, so that longer-term temporal patterns such as seasonal variation could only be assessed indirectly (Section 5.4).

6. CONCLUSION

This study presented a systematic comparison of seven models for predicting the energy consumption of 5G base stations on the ITU 5G Energy Consumption Modelling dataset, under a station-grouped cross-validation protocol. The main findings are as follows. (i) Gradient boosting (LightGBM) achieved the best overall accuracy among the seven models (R^2 = 0.910, WMAPE = 10.22%), and the multilayer perceptron matched it in R^2, with station-clustered bootstrap intervals confirming that the two are statistically indistinguishable in the original station partition (further partitions revealed a small but consistent advantage for LightGBM; Sections 4.5–4.6); however, LightGBM required roughly twenty times less training time, making it the most favourable accuracy–cost trade-off. Existing techniques re-implemented under the identical protocol confirmed this assessment: the most accurate of them, a published gradient-boosting–neural-network ensemble, improved on LightGBM by only 0.15 percentage points in WMAPE at about 30 times the training time, and an analytical load-dependent power model was clearly less accurate. (ii) Feature-importance

CC: BY-NC-ND

analysis showed that energy consumption is governed predominantly by static hardware attributes and their interaction with traffic load, rather than by traffic alone. (iii) Sparsely activated energy-saving signals destabilize linear models, producing physically impossible predictions; aggregating these signals and clipping predictions at zero resolves the issue. (iv) A marked hardware distribution shift exists between the training and evaluation partitions, which limits the scope of in-distribution evaluation and defines the central direction for future work: quantifying robustness under distribution shift using independently collected operational data, and extending the temporal scope beyond a single week. (v) The lower errors reported in the literature for this dataset (WMAPE of about 4–5%) are reproduced when station-specific history is available; they therefore characterize stations with measurement history rather than new deployments. (vi) Accuracy was stable across the observed days and under out-of-time validation, but tree ensembles underestimated consumption when traffic exceeded the training range, so that year-round use requires periodic recalibration or hybrid models with a linear component, and the validation of the findings on multi-season operational data remains essential.

CODE AND DATA AVAILABILITY

The complete preprocessing, training, evaluation, and analysis code, together with all aggregate result files and figures, is openly archived on Zenodo (DOI: https://doi.org/10.5281/zenodo.23238729). The archive also contains the scripts of the additional analyses (Sections 4.5–4.7) and the station-to-fold assignment used in them. The ITU 5G Energy Consumption dataset is subject to the distribution terms of the ITU AI/ML in 5G Challenge and is therefore not redistributed; the archive includes download instructions and a deterministic preprocessing pipeline that regenerates all intermediate files from the original data. The data are distributed by the ITU AI for Good challenge platform upon registration [3], and no DOI has been assigned to them by the provider. To identify the exact data used here persistently, the snapshot of the challenge data published in the ITU challenge repository is referenced through its Software Heritage persistent identifier, swh:1:cnt:c529efd4080f5b7302af1fe7ec2d343dcbfe6e89 [30], and the SHA-256 checksums of the five data files are listed in the Zenodo archive, so that any copy obtained from the provider can be verified.

Peer-review: External, Independent.

Acknowledgements:
The authors thank the International Telecommunication Union (ITU) and Huawei for making the 5G base station energy consumption dataset available through the ITU AI/ML in 5G Challenge, and the anonymous reviewers for their constructive comments, which improved this article.

Declarations:

1. Statement of Originality:
This work is original.

2. Author Contributions:
Concept: RE; ***Conceptualization:*** RE; ***Literature Search:*** RE; ***Data Collection:*** RE; ***Data Processing:*** RE; ***Analysis:*** RE; ***Writing – original draft:*** RE, BAİ; ***Writing – review & editing:*** RE, BAİ.

3. Ethics approval:
Not applicable.

4. Funding/Support:
This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

5. Competing Interests:
The authors declare no competing interests.

6. GenAI Usage Statement:

CC: BY-NC-ND

During the preparation of this work the author used Claude in order to revise the language and structure of the text. After using this tool, the author reviewed and edited the content as needed and takes full responsibility for the content of the published article.

7. Sustainable Development Goals:

REFERENCES

[1] Fowdur, T. P., Doorgakant, B. (2023). A review of machine learning techniques for enhanced energy efficient 5G and 6G communications. Engineering Applications of Artificial Intelligence, 122, 106032.

[2] Piovesan, N., López-Pérez, D., De Domenico, A., Geng, X., Bao, H., Debbah, M. (2022). Machine learning and analytical power consumption models for 5G base stations. IEEE Communications Magazine, 60, 56–62.

[3] International Telecommunication Union (2023). AI/ML for 5G-Energy Consumption Modelling by ITU AI/ML in 5G Challenge. ITU AI for Good. Available: https://zindi.africa/competitions/aiml-for-5g-energy-consumption-modelling [Accessed: 8 October 2026].

[4] Chen, T., Wang, Y., Chen, H., Zhao, Z., Li, X., Piovesan, N., Zhu, G., Shi, Q. (2024). Modelling the 5G energy consumption using real-world data: Energy fingerprint is all you need. arXiv preprint arXiv:2406.16929.

[5] Thantharate, A., Tondwalkar, A. V., Beard, C., Kwasinski, A. (2022). ECO6G: Energy and cost analysis for network slicing deployment in beyond 5G networks. Sensors, 22, 8614.

[6] Makroz, I., Nafidi, A., Gutiérrez-Sánchez, R., Aourik, S. (2024). A machine learning approach for energy consumption in 5G networks. ITU Journal on Future and Evolving Technologies, 5(4), 447–457.

[7] Grinsztajn, L., Oyallon, E., Varoquaux, G. (2022). Why do tree-based models still outperform deep learning on typical tabular data? Advances in Neural Information Processing Systems, 35, 507–520.

[8] Breiman, L. (2001). Random forests. Machine Learning, 45, 5–32.

[9] Friedman, J. H. (2001). Greedy function approximation: a gradient boosting machine. Annals of Statistics, 29, 1189–1232.

[10] Chen, T., Guestrin, C. (2016). XGBoost: A scalable tree boosting system. Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 785–794.

[11] Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., Liu, T.-Y. (2017). LightGBM: A highly efficient gradient boosting decision tree. Advances in Neural Information Processing Systems, 30.

CC: BY-NC-ND

[12] Arik, S. Ö., Pfister, T. (2021). TabNet: Attentive interpretable tabular learning. Proceedings of the AAAI Conference on Artificial Intelligence, 35, 6679–6687.

[13] Gorishniy, Y., Rubachev, I., Khrulkov, V., Babenko, A. (2021). Revisiting deep learning models for tabular data. Advances in Neural Information Processing Systems, 34, 18932–18943.

[14] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, 12, 2825–2830.

[15] Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Köpf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., Chintala, S. (2019). PyTorch: An imperative style, high-performance deep learning library. Advances in Neural Information Processing Systems, 32, 8024–8035.

[16] Ioffe, S., Szegedy, C. (2015). Batch normalization: accelerating deep network training by reducing internal covariate shift. International Conference on Machine Learning (ICML), 448–456.

[17] Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. Journal of Machine Learning Research, 15, 1929–1958.

[18] Kingma, D. P., Ba, J. (2015). Adam: A method for stochastic optimization. International Conference on Learning Representations (ICLR).

[19] Hollmann, N., Müller, S., Eggensperger, K., Hutter, F. (2023). TabPFN: A transformer that solves small tabular classification problems in a second. International Conference on Learning Representations (ICLR).

[20] López-Pérez, D., De Domenico, A., Piovesan, N., Xinli, G., Bao, H., Qitao, S., Debbah, M. (2022). A survey on 5G radio access network energy efficiency: Massive MIMO, lean carrier design, sleep modes, and machine learning. IEEE Communications Surveys & Tutorials, 24(1), 653–697.

[21] Zimmermann, R. S. (2024). Hybrid-boosted model with an approach inspired by a mixture of experts for 5G energy consumption. ITU Journal on Future and Evolving Technologies, 5(4), 493–502.

[22] Kapoor, S., Narayanan, A. (2023). Leakage and the reproducibility crisis in machine-learning-based science. Patterns, 4(9), 100804.

[23] Auer, G., Giannini, V., Desset, C., Godor, I., Skillermark, P., Olsson, M., Imran, M. A., Sabella, D., Gonzalez, M. J., Blume, O., Fehske, A. (2011). How much energy is needed to run a wireless network? IEEE Wireless Communications, 18(5), 40–49.

[24] Debaillie, B., Desset, C., Louagie, F. (2015). A flexible and future-proof power model for cellular base stations. IEEE 81st Vehicular Technology Conference (VTC Spring), 1–7.

[25] Piovesan, N., López-Pérez, D., De Domenico, A., Geng, X., Bao, H. (2023). Power consumption modeling of 5G multi-carrier base stations: A machine learning approach. IEEE International Conference on Communications (ICC), 3633–3638.

[26] Lorincz, J., Garma, T., Petrović, G. (2012). Measurements and modelling of base station power consumption under real traffic loads. Sensors, 12(4), 4281–4310.

[27] Spagnuolo, A., Petraglia, A., Vetromile, C., Formosi, R., Lubritto, C. (2015). Monitoring and optimization of energy consumption of base transceiver stations. Energy, 81, 286–293.

[28] Prokhorenkova, L., Gusev, G., Vorobev, A., Dorogush, A. V., Gulin, A. (2018). CatBoost: unbiased boosting with categorical features. Advances in Neural Information Processing Systems, 31.

CC: BY-NC-ND

[29] Shwartz-Ziv, R., Armon, A. (2022). Tabular data: Deep learning is not all you need. Information Fusion, 81, 84–90.

[30] ITU AI/ML in 5G Challenge (2024). 5G base station energy consumption dataset (BSinfo, CLdata, ECdata, estimation and prediction files), archived snapshot. Software Heritage persistent identifier swh:1:cnt:c529efd4080f5b7302af1fe7ec2d343dcbfe6e89 (revision swh:1:rev:0e5e768d1a58f665bb3121ce4cb83080990d6a78). Available: https://archive.softwareheritage.org/browse/revision/0e5e768d1a58f665bb3121ce4cb83080990d6a78/

[31] Roberts, D. R., Bahn, V., Ciuti, S., Boyce, M. S., Elith, J., Guillera-Arroita, G., Hauenstein, S., Lahoz-Monfort, J. J., Schröder, B., Thuiller, W., Warton, D. I., Wintle, B. A., Hartig, F., Dormann, C. F. (2017). Cross-validation strategies for data with temporal, spatial, hierarchical, or phylogenetic structure. Ecography, 40(8), 913–929.

[32] Field, C. A., Welsh, A. H. (2007). Bootstrapping clustered data. Journal of the Royal Statistical Society Series B: Statistical Methodology, 69(3), 369–390.

[33] Demšar, J. (2006). Statistical comparisons of classifiers over multiple data sets. Journal of Machine Learning Research, 7, 1–30.

[34] Bergmeir, C., Benítez, J. M. (2012). On the use of cross-validation for time series predictor evaluation. Information Sciences, 191, 192–213.

[35] Xu, F., Lin, Y., Huang, J., Wu, D., Shi, H., Song, J., Li, Y. (2016). Big data driven mobile traffic understanding and forecasting: A time series approach. IEEE Transactions on Services Computing, 9(5), 796–805.

[36] Zhang, M., Fu, H., Li, Y., Chen, S. (2019). Understanding urban dynamics from massive mobile traffic data. IEEE Transactions on Big Data, 5(2), 266–278.

[37] Quiñonero-Candela, J., Sugiyama, M., Schwaighofer, A., Lawrence, N. D. (2009). Dataset Shift in Machine Learning. MIT Press, Cambridge, MA.

CC: BY-NC-ND