IISE Data Analytics Challenge Report

Group Name: CMUGridElfs

Baoni Li *, Shuyi Chen*, Shixiang Zhu*, Wenbin Zhou*

Carnegie Mellon University

October 5, 2025

*Equal contribution. Authors listed in alphabetical order.

1 Introduction

Load forecasting is a vital area of study in the energy sector, as it underpins the reliable and efficient operation of power systems. By predicting future electricity demand over short, medium, or long-term horizons, load forecasting enables utility companies and grid operators to balance supply and demand, plan infrastructure investments, and integrate renewable energy sources effectively. As the power grid evolves with increasing penetration of distributed energy resources and electrification of sectors like transportation, accurate load forecasting becomes even more critical. Studying load forecasting is essential for ensuring energy reliability, reducing costs, and supporting the transition toward a more sustainable and resilient energy future [4].

To produce accurate load forecasts, temperature and global horizontal irradiance (GHI) are two powerful predictors that can be used. Temperature affects electricity usage primarily through heating and cooling needs [15]. Studies have demonstrated a clear correlation between temperature deviations from a comfort zone and increased electrical demand. For example, one study indicates that a 2°C rise in temperature can cause average daily peak demand to increase from 118.6% to 122.5% of the historic daily mean [12]. Another study found that in France, electricity demand increases by approximately 7% for each degree Celsius rise in temperature during daytime hours [11]; GHI measures the total shortwave radiation received per unit area by a horizontal surface from both direct sunlight and diffuse sky radiation. GHI influences ambient temperature and, consequently, electricity demand. Higher GHI levels typically lead to increased temperatures, which can elevate the use of air conditioning systems, thereby raising electricity consumption. On the other hand, for regions installed with solar panel penetration rates such as California, higher GHI may alleviate the load burden by increasing solar electricity generation [16].

Given the complex interactions between environmental factors and electricity demand, it is crucial to develop a robust load forecasting model that can effectively leverage these insights. In this report, prepared for the 2025 PG&E Data Analytics Challenge, we present a novel machine learning approach designed to forecast annual electricity load for an undisclosed region within California. The model utilizes a rich set of environmental features, including temperature and GHI measurements collected from multiple, randomly distributed locations across the region. The dataset consists of two years of historical hourly load and environmental data, which we use for model training and validation. Our primary objective is to generate a high-fidelity, hourly load forecast for the subsequent year, providing actionable insights to support grid management and strategic planning.

For this purpose, we present several key findings and propose some novel methodological contributions in this report, which we summarize as follows:

- We introduce a bi-level ensemble framework that separates temporal pattern modeling from feature-driven load dynamics. By decoupling the time series structure from environmental predictors, this design improves the model's ability to capture both intrinsic temporal dependencies and exogenous influences on electricity demand.

- In addition, we develop a novel tri-resolution boosting strategy, which decomposes the forecasting task into monthly, weekly, and hourly components. Each tempo-

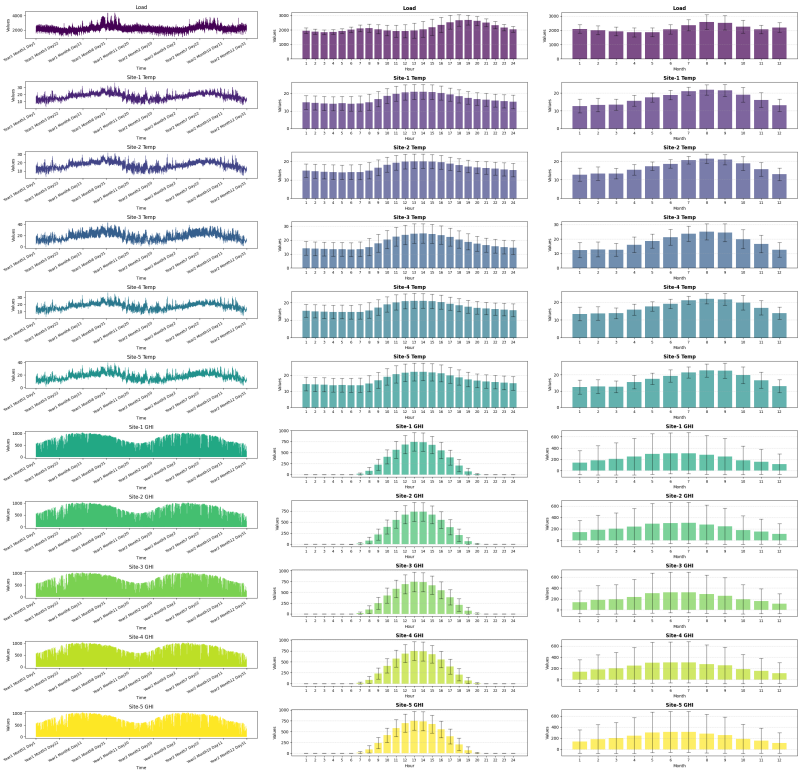
ral resolution is modeled using specialized methods such as ARIMAX, RNNs, Prophet, and XGBoost, allowing the model to effectively capture both coarse-grained seasonal patterns and fine-grained temporal variations based on environment predictors.

- We validate our proposed methods through numerical experiments on the dataset, demonstrating both their efficiency (requiring only a few seconds to generate final predictions) and their superior accuracy, consistently ranking first among baseline methods in our evaluation system.

Related Works Load forecasting has been extensively studied with numerous review papers have synthesized the prevailing trends and methodologies. For instance, [14] categorizes forecasting techniques based on underlying model types, [10] provides a comprehensive overview of electricity load forecasting methods, and [9] discusses forecasting approaches in the context of various drivers and application scenarios. Broadly, load forecasting methods can be grouped into three major categories: (1) regression-based approaches, including linear regression, multiple linear regression, and autoregressive models such as AR, ARIMA, and ARIMAX; (2) machine learning methods, including neural networks (NN), artificial neural networks (ANN), support vector machines (SVM), recurrent neural networks (RNN), and long short-term memory (LSTM) networks; and (3) other soft computing techniques.

Despite the maturity of the field, long-term load forecasting (LTLF) has received comparatively limited attention, accounting for only about 5% of the literature relative to short-term (STLF) and mid-term (MTLF) forecasting studies [10]. Existing LTLF methods include regression-based models and artificial neural networks (ANN) [5], multiple linear regression approaches [2], hybrid LSTM-based frameworks [6], and optimization-based techniques such as particle swarm optimization for parameter tuning [3].

Hourly long-term load forecasting, which is the focus of this work, introduces additional challenges compared to conventional LTLF. The need to produce high-resolution (hourly) forecasts over an extended time horizon exacerbates issues such as capturing multi-scale seasonality, increased data sparsity and noise, and model complexity. Traditional methods often struggle to balance computational efficiency with accuracy when applied to such fine-grained, year-long forecasts. Although prior studies have explored multiple linear regression models [7], LSTM-based methods [1], and knowledge-driven frameworks [8], these approaches frequently encounter limitations in fully capturing both the short-term fluctuations and the long-term trends required for reliable hourly LTLF. To address these challenges, we propose a bi-level ensemble framework that separates time series components from environmental predictor-based components, and further incorporate a tri-resolution ensemble structure to decompose the forecasting task into monthly, weekly, and hourly layers. These multi-resolution strategies enhance forecasting accuracy while significantly improving computational efficiency for long-horizon, high-resolution predictions.



(a) All (b) Daily (c) Monthly

Figure 1: Exploratory data visualization

2 Data Description and Analysis

This section provides an overview and exploratory analysis of the training data used in our study. The dataset comprises 17,544 entries of hourly time series records over a two-year period, capturing temperature and global horizontal irradiance (GHI) from five sites located near the target area. Figure 1a presents the full dataset. A preliminary inspection reveals pronounced seasonal patterns: load demand is significantly higher during the summer months (July to October) and lower during the rest of the year. Similar seasonal trends are also evident in the temperature and GHI time series. These observations underscore the importance of incorporating seasonality into the modeling framework.

A key observation from the dataset is the temporal misalignment between load demand and its environmental drivers. Specifically, while load demand typically peaks during the morning (6–9) and evening (16–21) hours, temperature and GHI reach their maxima around midday to early afternoon (12–16), as illustrated in the average daily patterns (Figure 1b). This temporal shift suggests a complex, likely nonlinear relationship between load, temperature, and GHI at the hourly scale, highlighting the need for advanced modeling approaches, such as machine learning and deep learning, capable of capturing these dynamics. A similar trend is observed at the monthly level (Figure 1c): GHI peaks around June, whereas both temperature and load reach their highest levels later in the summer.

These observations motivate the design of a hierarchical modeling framework capable

of capturing multi-scale seasonality and nonlinear feature-load interactions. In the next section, we present our proposed bi-level and tri-resolution ensemble models to address these challenges.

3 Methodology

To address the challenges identified in Section 2, we propose and compare two hierachical modeling frameworks: a bi-level ensemble model and a tri-resolution boosting model. Both models are designed to capture multi-scale seasonality and nonlinear dependencies between environmental covariates and electricity demand, but they differ in how they decompose temporal patterns.

3.1 Bi-level Model: Decoupling Temporal and Feature-driven Components

The bi-level model decomposes the forecasting task into two complementary components: temporal patterns and feature-driven dynamics, using a residual learning framework.

Model Details First, to capture temporal patterns, we fit a traditional time series model (e.g., ARIMA or Prophet) to the load series Y_t. This component focuses on modeling long-term trends and seasonality based purely on historical load observations.

Next, we address feature-driven dynamics by modeling the residuals left unexplained by the time series model. Specifically, we compute the residuals as:

$$R_t = Y_t - \widehat{Y}_t^{\mathrm{TS}}, \tag{1}$$

where $\widehat{Y}_t^{\mathrm{TS}}$ is the prediction from the temporal model. We then train a Feature-driven Residual Model on R_t using a machine learning method such as XGBoost. This model takes environmental covariates X_t (e.g., temperature, GHI) as inputs, capturing the nonlinear effects of external factors on the residual fluctuations.

This bi-level structure ensures that the temporal and feature-driven components are optimized independently, enabling each model to leverage its respective strengths—such as capturing seasonality and trends with the time series model and modeling complex, nonlinear relationships with the machine learning model.

3.2 Tri-resolution Boosting Model

The tri-resolution boosting model builds upon the bi-level framework by further decomposing the forecasting task into three hierarchical temporal resolutions: monthly, weekly, and hourly. The model sequentially fits specialized predictors at each resolution using a boosting strategy [13], where the residuals from coarser resolutions are passed to finer resolutions. Simple statistical models (e.g., ARIMAX) are applied to low-frequency (monthly) components, while more flexible machine learning models (e.g., RNN and XGBoost) are used to capture high-frequency (hourly) variations.

The motivation arises from our data analysis, which reveals strong seasonality at the aggregated level and rich, dynamic patterns at the fine-grained hourly level. This dual

nature poses challenges for both statistical and machine learning models: statistical models, such as traditional AR models, are typically well-suited for capturing seasonality but lack the complexity to model fine-scale dynamics; conversely, ML models are often overparameterized for the sparse seasonal patterns, making them better suited for high-resolution variations but less efficient at capturing broader seasonal trends. Moreover, modeling seasonality at an hourly resolution over a two-year period is computationally intensive—e.g., SARIMAX can take several hours to converge.

Algorithm 1 Tri-resolution Boosting Prediction Model

1: **Input:** Number of samples N; Covariates matrix $X \in \mathbb{R}^{N \times C}$ and feature vector $Y \in \mathbb{R}^N$; Timestamp metadata.
2: De-trend the data at the monthly and hourly resolution using simple mean estimates.
3: Compute hour-to-month and hour-to-week projection zero-one matrices:

$$P_1 \in \{0,1\}^{N \times 24}, \quad P_2 \in \{0,1\}^{N \times W}$$

using the timestamp metadata.
4: Fit first-level predictor (e.g., ARIMAX) on monthly data

$$(P_1^\top X, P_1^\top Y).$$

Denote the fitted values as: $\widehat{Y}_1 \in \mathbb{R}^{24}$.
5: Fit second-level predictor (e.g., RNN/Prophet) on weekly data:

$$(P_2^\top X, P_2^\top Y - (P_2^\top P_1)\widehat{Y}_1).$$

Denote the fitted values as: $\widehat{Y}_2 \in \mathbb{R}^W$.
6: Fit third-level predictor (e.g., XGBoost) on hourly (original) data:

$$(X, Y - P_2\widehat{Y}_2 - P_1\widehat{Y}_1).$$

Denote the fitted values as: $\widehat{Y}_3 \in \mathbb{R}^N$.
7: Output prediction value:

$$\widehat{Y} = \widehat{Y}_3 + P_2\widehat{Y}_2 + P_1\widehat{Y}_1.$$

Model Details The tri-resolution boosting models posit that the data generation process can be decomposed as three simple additive terms:

$$Y_t = Y_{t(\text{month})} + Y_{t(\text{date})} + Y_{t(\text{hour})} + \epsilon_t, \tag{2}$$

where we use t to denote the current timestamp, and $t(\cdot)$ denotes the month/date/hour extracted from the current timestamp. ϵ_t is an exogenous error term which we assume to be independent from the output variable Y_t and its decomposed parts. The three decomposed parts will each be independently captured by a time series model, which we refer to as the monthly, daily, and hourly model. Specifically, each model aims to capture the residual of the previous model (*e.g.,* the daily model aims to capture the

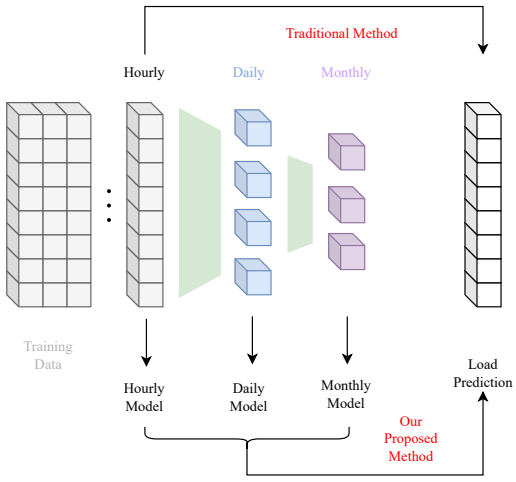


Figure 2: Illustration of our proposed model pipeline. The key novelty lies in decomposing the data into three temporal resolutions, each modeled separately and aggregated via a boosting-style approach to achieve efficient training and accurate load prediction.

residual $Y_t - Y_{t(\text{month})}$) and summed up in the final prediction. Such a structure ensures that each model can utilize their distinct advantage to capture the dynamics that were omitted or unable to be captured by the previous models, therefore achieving a better modeling outcome.

Efficiency We improve the efficiency of this framework by further proposing to break down the training process into multi-stage sequential training. Each stage corresponds to a by using linear projections to align the temporal resolutions. This enables efficient training and less architectural overhaul of the deployed time series models, facilitating easy implementation in practice. This enables our model to be able to be integrated with the existing package implementations of training and prediction. Note that without this technique, training the model would require a new global loss function for training three models jointly according to (2), which is undesirable. The pseudocode of the entire algorithm is provided in Algorithm 1.

4 Numerical Experiment

In this section, we conduct holistic numerical experiments to evaluate the performance of different existing methods and our proposed method with their combinations. These performance results are analyzed and used to determine the final prediction model for delivering the results.

4.1 Performance Evaluation

Hardware Specifications All experiments are conducted on Google Colab (free tier) with up to 2-core 2.20GHz Intel Xeon CPUs, 12.7 GB RAM, and 50 GB disk.

Evaluation Metric We partition the training data into a training split (80%) and a validation split (20%). Models are compared based on their mean squared error (MSE) on the validation split. For models requiring hyperparameter tuning, such as the lag window size for XGBoost or the number of layers and input sequence length for RNNs, a 20% hold-out tuninng set from the training split is used. After tuning, the selected model is retrained on the entire training dataset before generating predictions on the test set.

Results and Findings Among all evaluated methods, our proposed tri-level boosting model, with all components implemented using XGBoost, achieves the best performance, yielding the lowest validation mean squared error (MSE) of approximately 43.04. This significantly outperforms all other baselines, including the runner-up bi-level approach using RNN and XGBoost Half model as reported in Table 1. These results underscore the superior capability of our approach in capturing temporal dependencies across multiple resolutions. Figure 3 provides additional visual comparisons between the baseline models and our proposed approach, further supporting the overall effectiveness of our model in both in-sample and out-of-sample settings.

Table 1: Comparison of Validation MSE for All Methods

Single Models	
`Prophet`	402.36
`XGBoost Full`	271.03
`RNN`	214.51
`XGBoost Half`	193.15
Bi-level Model	
`Prophet + XGBoost Half`	358.73
`XGBoost Half + RNN`	190.59
`XGBoost Half + Prophet`	178.91
`RNN + XGBoost Half`	<u>58.44</u>
Tri-resolution Boosting	
`Tri-level Boosting`	**43.04**

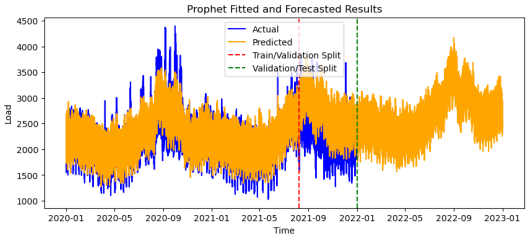
Hyperparameter Tuning Table 2 illustrates our hyperparameter tuning process for the XGBoost model. The XGBoost Half model—trained only on exogenous covariates—outperforms the Full model, which includes lagged historical load values. This is expected, as the Full model's rolling prediction strategy recursively feeds its own forecasts into future inputs, leading to cumulative error. These results support our choice of the Half model for more stable and reliable forecasting. The table also shows that a lag of 11 yields the lowest validation MSE on the tuning set; thus, we use a lag of 12 for all XGBoost Half models.

5 Conclusion

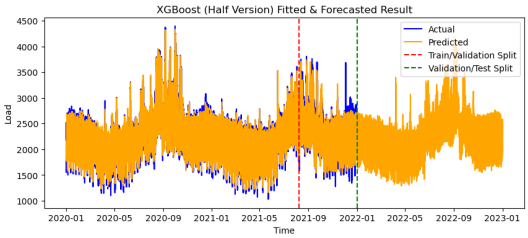
We presented two hierarchical models for long-term, high-resolution load forecasting: a bi-level ensemble model and a tri-resolution boosting model. The bi-level model separates temporal trends from feature-driven dynamics, while the tri-resolution model further decomposes patterns into monthly, weekly, and hourly components. Our experiments showed that the tri-resolution boosting model outperforms baseline methods, achieving superior accuracy and computational efficiency. By effectively capturing multi-scale seasonality and nonlinear relationships, our approach provides a robust and scalable solution for electricity load forecasting.

Table 2: Tuning Resutls for Different `XGBoost` Lag Window Sizes in the Tuninng Set

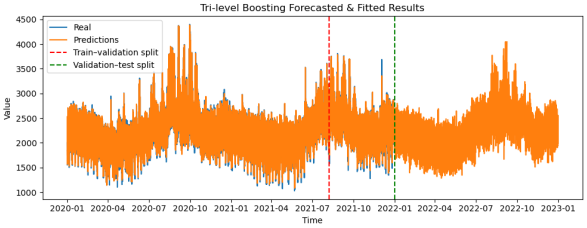
Lag (Window)	XGBoost (Half)	XGBoost (Full)
1	267.44	379.60
3	182.33	744.31
6	177.12	618.35
8	169.56	598.86
10	168.31	529.97
11	168.27	480.81
12	168.63	485.43
15	172.99	476.45
20	171.71	518.07
24	172.09	506.34
30	174.49	498.60



(a) `Prophet` (MSE: 402.36)



(b) `XGBoost Half` (MSE: 193.15)



(c) `Tri-level Boosting` (MSE: 43.04)

Figure 3: Comparison of model predictions and validation MSEs.

References

[1] Rahul Kumar Agrawal, Frankle Muchahary, and Madan Mohan Tripathi. Long term load forecasting with hourly predictions based on long-short-term-memory networks. In *2018 IEEE Texas Power and Energy Conference (TPEC)*, pages 1–6. IEEE, 2018.

[2] HM Al-Hamadi and SA Soliman. Long-term/mid-term electric load forecasting based on short-term correlation and annual growth. *Electric power systems research*, 74(3):353–361, 2005.

[3] MR AlRashidi and KM El-Naggar. Long term electric load forecasting based on particle swarm optimization. *Applied Energy*, 87(1):320–326, 2010.

[4] Shuyi Chen, Ferdinando Fioretto, Feng Qiu, and Shixiang Zhu. Global-decision-focused neural odes for proactive grid resilience management. *arXiv preprint arXiv:2502.18321*, 2025.

[5] Hossein Daneshi, Mohammad Shahidehpour, and Azim Lotfjou Choobbari. Long-term load forecasting in electricity market. In *2008 IEEE international conference on electro/information technology*, pages 395–400. IEEE, 2008.

[6] Ming Dong and Lukas Grumbach. A hybrid distribution feeder long-term load forecasting method based on sequence prediction. *IEEE Transactions on Smart Grid*, 11(1):470–482, 2019.

[7] Tao Hong, Jason Wilson, and Jingrui Xie. Long term probabilistic load forecasting and normalization with hourly information. *IEEE Transactions on Smart Grid*, 5(1):456–462, 2013.

[8] M Rostam Niakan Kalhori, I Taheri Emami, F Fallahi, and M Tabarzadi. A data-driven knowledge-based system with reasoning under uncertain evidence for regional long-term hourly load forecasting. *Applied Energy*, 314:118975, 2022.

[9] KB Lindberg, P Seljom, H Madsen, D Fischer, and M Korpås. Long-term electricity load forecasting: Current and future trends. *Utilities Policy*, 58:102–119, 2019.

[10] Isaac Kofi Nti, Moses Teimeh, Owusu Nyarko-Boateng, and Adebayo Felix Adekoya. Electricity load forecasting: a systematic review. *Journal of Electrical Systems and Information Technology*, 7:1–19, 2020.

[11] Yaju Rajbhandari, Anup Marahatta, Bishal Ghimire, Ashish Shrestha, Anand Gachhadar, Anup Thapa, Kamal Chapagain, and Petr Korba. Impact study of temperature on the time series electricity demand of urban nepal for short-term load forecasting. *Applied System Innovation*, 4(3):43, 2021.

[12] Michael J Roberts, Sisi Zhang, Eleanor Yuan, James Jones, and Matthias Fripp. Using temperature sensitivity to estimate shiftable electricity demand. *Iscience*, 25(9), 2022.

[13] Robert E Schapire. The boosting approach to machine learning: An overview. *Nonlinear estimation and classification*, pages 149–171, 2003.

[14] Arunesh Kumar Singh, S Khatoon, Md Muazzam, Devendra Kumar Chaturvedi, et al. Load forecasting techniques and methodologies: A review. In *2012 2nd International Conference on Power, Control and Embedded Systems*, pages 1–10. IEEE, 2012.

[15] Patrick Sullivan, Jesse Colman, and Eric Kalendra. Predicting the response of electricity load to climate change. Technical report, National Renewable Energy Lab.(NREL), Golden, CO (United States), 2015.

[16] Wenbin Zhou, Shixiang Zhu, Feng Qiu, and Xuan Wu. Hierarchical spatio-temporal uncertainty quantification for distributed energy adoption. *arXiv preprint arXiv:2411.12193*, 2024.