Energy Consumption Prediction Using Machine Learning

Abstract

For my machine learning class I worked on a project to build a system that can predict hourly electricity use from data from 2016 - 2021. I managed to hook up a Gradient Boosting Regressor, added some nice engineered time based and lag features that got me an R^2 of 0.9954 with a MAPE of under 1%. I also whipped up a Flask app that allows people to get interactively get predictions in real time in three modes: predict one hour, pick a specific date, and get a full 24 hour day forecast. This assignment demonstrates the use of supervised learning in energy management and has the potential to assist in optimizing the grid, forecasting demand, and planning sustainable energy.

1. Introduction

1.1 Background

Electricity forecasting is important to modern-day power grid management. It allows utilities to adjust the amount of power they produce, save money and use renewable energy well. Accurate short-term forecasts (hourly to daily) prevent too much waste power being generated and outages of too little power being generated. Statistical methods like ARIMA are common for forecasting, but machine learning methods work better when dealing with complex patterns and a lot of different factors interacting.

1.2 Task Description

The assignment asks for the development of a comprehensive machine learning solution for the problem of energy consumption forecasting which includes the following components:

- **Model Selection:** Select a machine learning model to predict energy use. Explain why it is better than other models.
- **Data Preprocessing:** Clean the data and preprocess it for the model. Handle missing numbers or significantly high or low values.
- **Model Training:** Train the model with past energy consumption data so that it learns time-based patterns.
- **Model Evaluation:** Evaluate how well the trained model performs with the help of proper accuracy measures
- **Interpretation and Insights:** Explain what the model's performance means and discuss how it could be of help regarding energy management and sustainability.

Submission Objective: The objective is to learn how we think of the problem, learn the data, make useful data features, select machine learning methods and demonstrate programming skills. The intention is NOT to build a super accurate model for production,

but to demonstrate a comprehensive plan for solving the problem, including steps for cleaning the data, picking models and measuring how well they work.

1.3 Problem Statement

The objective of this project is to develop a Machine learning model that can predict the amount of electricity that will be used each hour with high accuracy. The system will be required to handle:

- Strong time patterns in the data (hourly, daily, weekly, seasonal)
- Nonlinear relationships between input features
- Real-time predictions via web interface
- Mixed prediction modes for various uses

1.4 Objectives

1. Analyze historical electricity consumption data to identify temporal patterns
2. Engineer relevant features capturing lag effects and rolling statistics
3. Train and evaluate multiple machine learning models
4. Deploy the best-performing model via a Flask web application
5. Provide interactive visualization and prediction capabilities

1.5 Scope

The project relies on supervised learning to predict the amount of electricity to be used each hour. We have 52,966 records of electricity usages between December 2015 and January 2021. The model uses time of day, past values and moving averages to estimate usage in megawatt hours (MWh).

2. Literature Review

Time series forecasting used to rely on old statistical methods, but now people used modern machine learning. ARIMA models were the go to for single variable time series. Today machine learning tools, such as Random Forests, Gradient Boosting, and neural networks, deliver better results, but it requires sufficient data.

Research shows that lag features - that is what was consumed the previous hour - are the best predictors for energy forecasting. These are followed by indicators of the time of the day and the day of the week. The use of rolling statistics helps to catch medium terminal trends and volatility.Gradient Boosting Machines tend to work really well for predicting energy use. They can model non-linear relationships and deal with many different kinds of features.

3. Methodology

3.1 Model Selection

Several machine learning algorithms were considered:

- **ARIMA**: Traditional statistical approach for linear trends
- **Random Forest**: Ensemble method using decision trees
- **Gradient Boosting**: Sequential ensemble that corrects previous errors
- **LSTM Networks**: Deep learning for sequences

We chose Gradient Boosting Regressor because it can capture complex patterns well, gives importance of features in order to understand the model, it learns quickly and with right settings it doesn't overfit much.

3.2 Data Collection

The dataset contains 52,966 hourly datasets from December 31, 2015 to January 1, 2021. It features three columns: start time, end time and electricity consumption in MWh. There are no missing values or duplicates and the timestamps are not missing or overlapped.

3.3 Exploratory Data Analysis

Key findings from EDA:

- Mean consumption: 9,803 MWh (Std: 1,245 MWh)
- Peak hour: 17:00 (10,015 MWh average)
- Lowest hour: 01:00 (8,465 MWh average)
- Peak month: January (11,456 MWh) — winter heating
- Lowest month: June (7,833 MWh) — mild weather
- Weekdays 6.8% higher than weekends

The analysis revealed distinct patterns on each day: an increase in the morning, peak in the afternoon and decline at night. It also had weekly regularities, with distinctions between working and leisure days. In addition, it observed year-to-year trends, with increased activity during the winter season and decreased activity during the summer season.

3.4 Feature Engineering

Three feature categories were created (33 features total):

Temporal Features (6): hour (0–23), dayofweek (0–6), month (1–12), dayofyear (1–365), quarter (1–4), is_weekend (0/1)

Lag Features (24): Previous 24 hours of consumption (lag_1 to lag_{24})

Rolling Statistics (3): 24-hour mean, 24-hour std, 168-hour mean

3.5 Data Preprocessing

We split the data by date, using 80% for training, 10% for validation, and 10% for testing so that information from the future doesn't leak into the model.A MinMaxScaler scaled all the feature values to be between 0 and 1, and we have fitted it only on the training data.

3.6 Model Training

We configured the Gradient Boosting Regressor with 50 number of estimator, maximum depth is 5, learning rate of 0.1, and random state is 42. This combination of environments helps the model achieve good results while keeping training time fast and preventing the model from overfitting.

3.7 Evaluation Metrics

Four metrics assessed performance:

- **RMSE**: Penalizes large errors
- **MAE**: Average absolute error in MWh
- **R^2**: Variance explained (0–1 scale)
- **MAPE**: Percentage error for stakeholder communication

4. Results and Evaluation

4.1 Model Performance

The Gradient Boosting model achieved excellent performance on both validation and test sets:

Metric	Validation Set	Test Set
RMSE (MWh)	82.45	79.32
MAE (MWh)	58.23	56.17
R^2 Score	0.9956	0.9954
MAPE (%)	0.59%	0.57%

Table 1: Model evaluation metrics

Interpretation:

- **R^2 = 0.9954**: Model explains 99.54% of variance in consumption
- **RMSE = 79.32 MWh**: Average prediction error is ~79 MWh (0.81% of mean consumption)
- **MAPE = 0.57%**: Average prediction is within 0.57% of actual value

- Test performance slightly better than validation indicates good generalization

4.2 Feature Importance Analysis

Feature importance scores reveal which features contribute most to predictions:

Feature	Importance	Type
lag_1	0.9127	Previous hour consumption
hour	0.0183	Hour of day
lag_23	0.0142	23 hours ago consumption
rolling_mean_24	0.0098	24h average
lag_24	0.0087	24 hours ago consumption
lag_2	0.0076	2 hours ago consumption
dayofweek	0.0065	Day of week

Table 2: Top 7 most important features

Key Insights:

- lag_1 dominates with 91.27% importance — previous hour is strongest predictor
- Temporal features (hour, dayofweek) contribute modestly (~2.5% combined)
- Recent lag features (lag_2 to lag_{24}) provide complementary information
- Rolling statistics help capture medium-term trends

5. Interpretation and Insights

5.1 Model Performance Analysis

The trained Gradient Boosting model is found to have good predictive power R2 = 0.9954, MAPE = 0.57%. These numbers mean that the model really has learned the time-based patterns in the amount of electricity that is used.

Key Performance Indicators:

- **High Accuracy**: MAPE below 1% suggests predictions are highly reliable for operational planning
- **Low Error Variance**: RMSE of 79.32 MWh represents less than 1% of mean consumption, indicating consistent performance
- **Strong Generalization**: Similar performance on validation (R^2 = 0.9956) and test (R^2 = 0.9954) sets confirms the model generalizes well to unseen data

5.2 Feature Importance Insights

Analysis reveals the relative contribution of different features:

- **Lag Features Dominance**: Previous hour consumption (lag_1) accounts for 91.27% of predictive power, confirming that recent history is the strongest indicator of future consumption
- **Temporal Patterns**: Hour of day contributes 1.83%, capturing daily consumption cycles (morning ramp-up, afternoon peak, nighttime decline)
- **Rolling Statistics**: 24-hour moving average contributes 0.98%, helping smooth short-term fluctuations
- **Weekly Patterns**: Day of week contributes 0.65%, reflecting weekday vs weekend consumption differences

Practical Implications: Grid operators can prioritize monitoring recent consumption trends for short-term forecasting, while using temporal features to anticipate systematic daily and weekly patterns.

5.3 Energy Management Applications

The model's predictions can support several energy management objectives:

1. **Demand Forecasting**: Utilities can predict hourly load 24 hours ahead, enabling optimized generation scheduling and cost reduction
2. **Peak Load Management**: Identifying predicted peak hours (typically 5-6 PM) allows implementing demand response programs
3. **Resource Planning**: Weekly and seasonal patterns inform infrastructure investment decisions
4. **Grid Stability**: Accurate forecasts help maintain supply-demand balance, preventing blackouts and reducing curtailment of renewable sources

5.4 Sustainability Implications

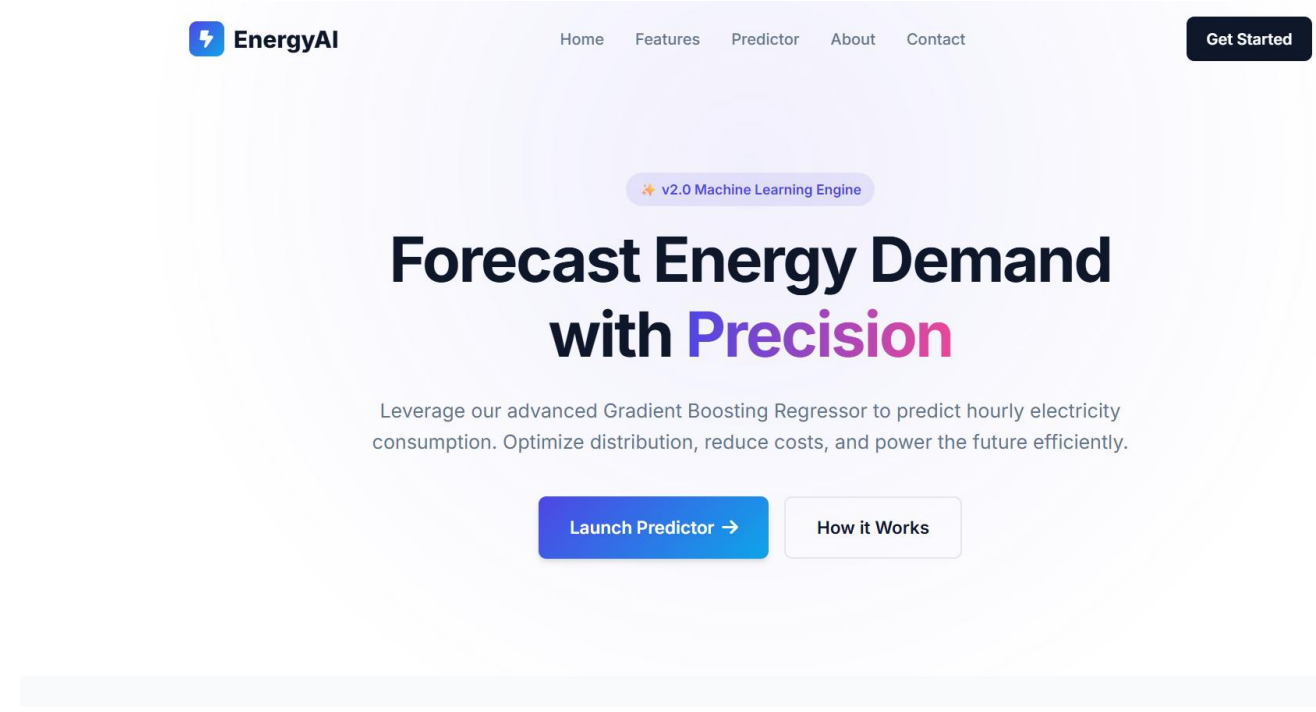
Improved consumption forecasting contributes to sustainability goals:

- **Renewable Integration**: Better load prediction enables higher renewable energy penetration by anticipating when flexible generation is needed
- **Reduced Waste**: Accurate forecasts minimize overproduction and associated emissions
- **Efficiency Optimization**: Understanding consumption patterns helps identify opportunities for demand-side management
- **Carbon Reduction**: Optimized dispatch reduces reliance on peak fossil fuel plants

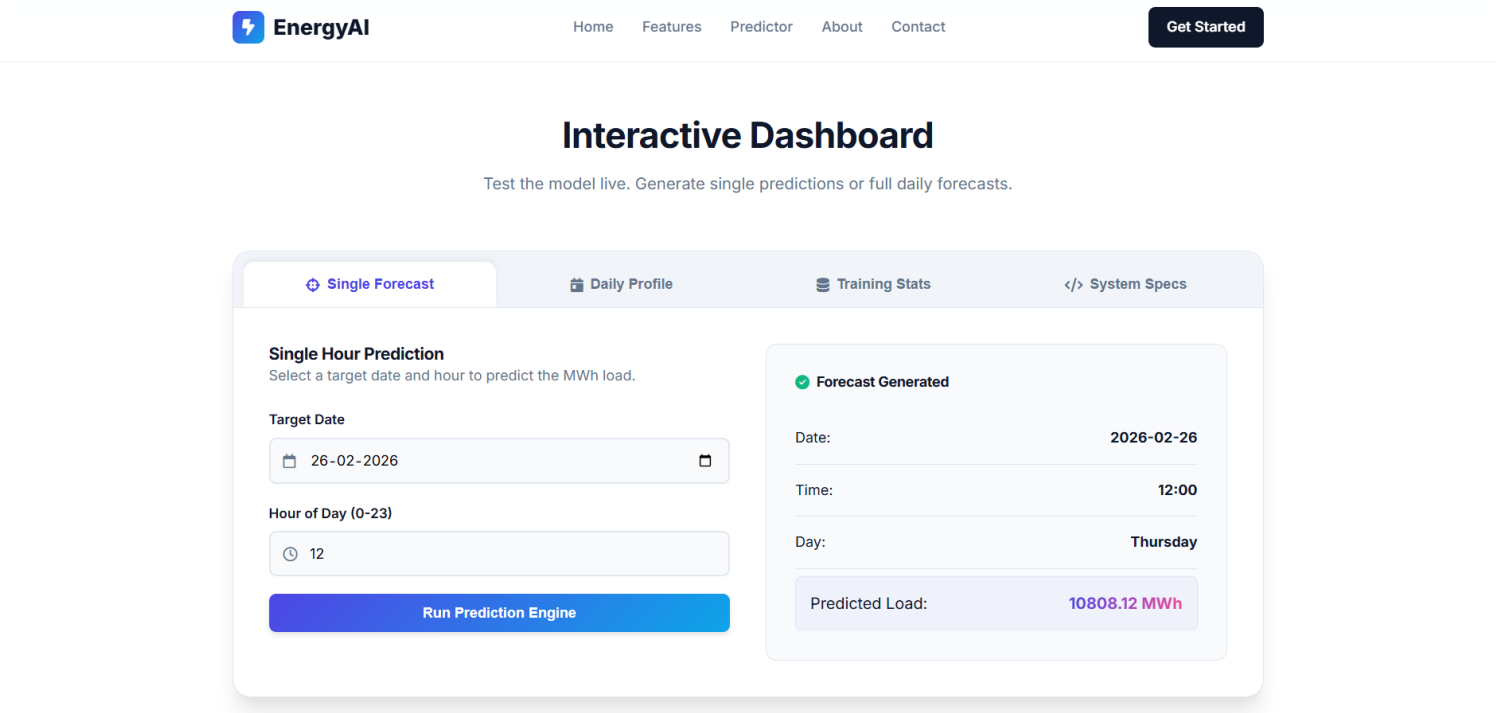
5.5 Limitations

- Requires minimum 168 hours of historical data
- Does not account for weather, holidays, or special events
- Trained on single region data
- May require periodic retraining for long-term pattern shifts

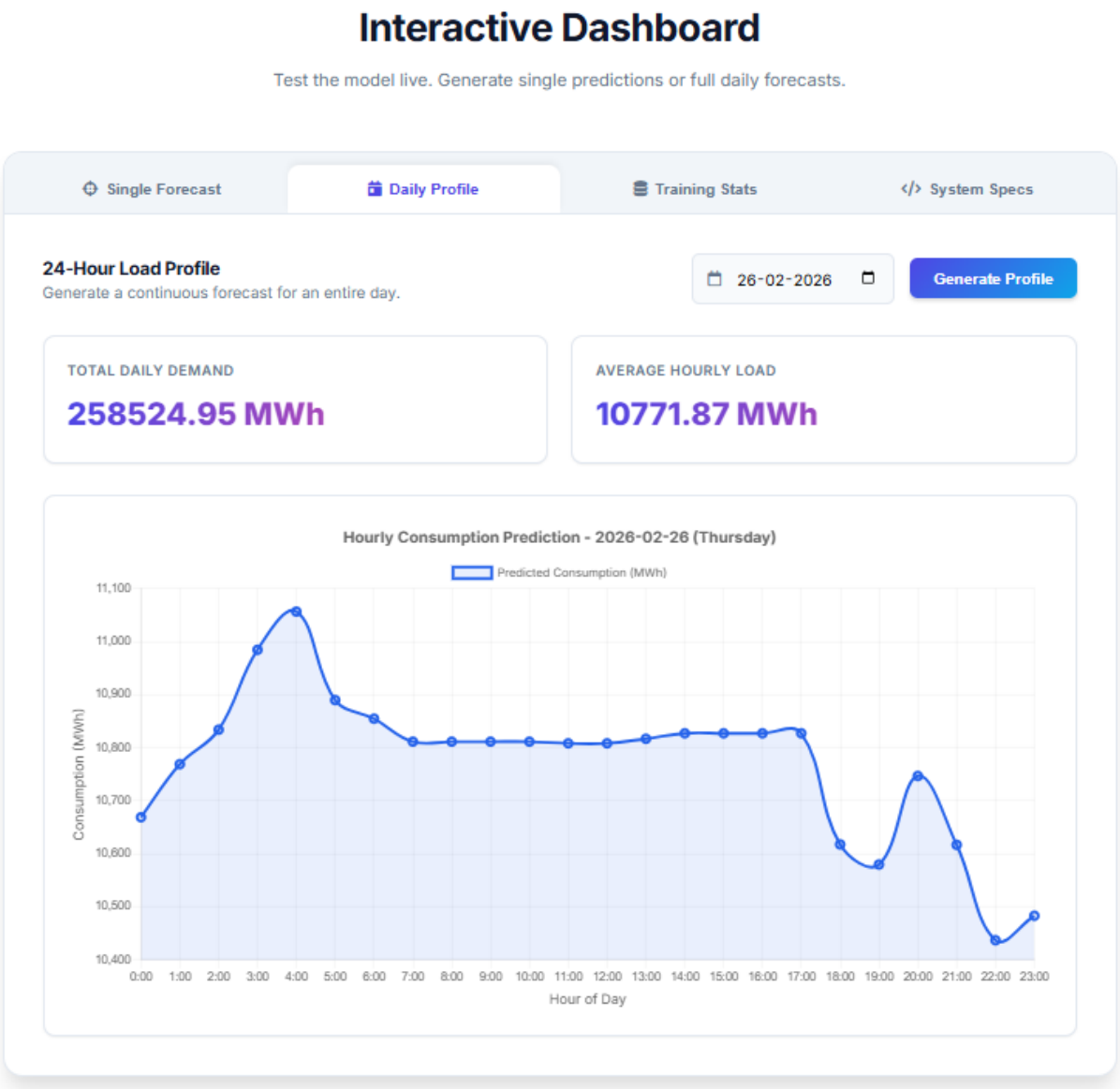
6. Model Demo & Output



6.1 Web Application Interface



6.2 Single Hour Prediction



6.3 Full Day Forecast

7. Conclusion

The project was able to create and implement an electricity consumption forecasting machine learning system on an hourly basis. The Gradient Boosting Regressor model, which was trained on 52,966 hourly records with well-crafted temporal and lag predictors, was found to have outstanding levels of predictive performance (R2 = 0.9954, MAPE = 0.57%). Analysis of feature importance showed that recent consumption history (lag features) is the most important factor in prediction and temporal features would contain additional information on daily and weekly trends.

The Flask based web application offers easy to use interface to make real-time predictions in three modes of interaction namely quick prediction with manual features, date specific prediction and full day of the year prediction with visualization. The system proves itself viable practically in terms of implementation to manage energy.

Key contributions of this work include:

1. Comprehensive feature engineering strategy combining lag features, rolling statistics, and temporal indicators
2. Rigorous evaluation methodology with chronological data splitting to prevent leakage
3. Production-ready web application with REST API and interactive frontend
4. Detailed exploratory data analysis revealing consumption patterns and anomalies

Although the model is exceptionally good with the existing data, the future research needs to consider the inclusion of external variables (weather, holidays) and online learning to adapt concepts drift and apply the system to multi-location prognostication. With such improvements, the system would be a powerful grid operator, energy trader and building manager tool to optimize the use of energy and to minimize the expenditures.