Energy Consumption Predictor

Machine Learning-Based Prediction System

Prepared By: Shivaji Puri

Date: 30 August 2025

1. Executive Summary

This project introduces a Machine Learning-powered application designed to predict energy consumption in buildings. The system combines a Flask backend, Linear Regression model, and Bootstrap frontend to deliver an easy-to-use tool. It helps residential, commercial, and industrial stakeholders optimize energy use, reduce costs, and support sustainable initiatives. By simulating multiple scenarios, clients can better plan infrastructure, manage operational costs, and adopt energy-efficient practices.

2. Project Overview

The Energy Consumption Predictor is a web application that forecasts energy usage based on building attributes and usage patterns. It considers factors such as square footage, number of occupants, appliance usage, average temperature, day type (weekday/weekend), and building type (residential, commercial, industrial). The backend is powered by a trained Linear Regression model, while the frontend provides an intuitive interface for entering inputs and visualizing results. The solution is designed for accessibility, scalability, and deployment on cloud or server environments.

3. Objectives

- Provide accurate and fast predictions of energy consumption.
- Enable users to test scenarios by adjusting input parameters.
- Assist organizations in energy optimization and cost savings.
- Promote sustainability by supporting efficient energy planning.

4. Problem Statement

Energy consumption estimation is challenging due to variations in occupancy, appliances, and environmental factors. Traditional manual methods are inaccurate and fail to capture real usage patterns. Two buildings of identical size may have drastically different consumption levels depending on how they are used. This leads to inefficiencies, unexpected costs, and wasted energy. A data-driven approach is necessary to provide reliable forecasting for better decision-making.

5. Scope & Goals

Scope:
- Predict consumption for residential, commercial, and industrial buildings.
- Provide predictions via a device-accessible web platform.

Goals:
- Make predictions reliable, interpretable, and user-friendly.
- Offer scenario-testing to guide client planning and investment.
- Ensure solution scalability for real-world deployment.

Results

- **Dataset Size** : 10,000 records
- **Train-Test Split** : 80/20
- **Model Used** : Linear Regression
- **R^2 Score** : 0.82 (Good Fit)
- **MAE (Mean Absolute Error)** : 12.5 units
- **RMSE (Root Mean Square Error)** : 18.3 units
-

6. Proposed Solution

The proposed solution integrates **data-driven modeling** with a **simple web interface** for predicting building energy consumption.

Key Features Considered

- Square Footage
- Number of Occupants
- Appliances Used
- Average Temperature
- Day of Week (Weekday/Weekend)
- Building Type (Residential / Commercial / Industrial)

Target Variable: Predicted Energy Consumption (units)

Workflow

1. **Data Preprocessing & Model Training**
   - Performed cleaning, encoding, and scaling on raw data.
   - Trained a **Linear Regression** model for energy consumption prediction.
2. **Model Storage**
   - Trained model saved as Linear_Model.pkl for efficient loading and predictions.
3. **Metadata Management**
   - Feature mappings and configurations stored in project_data.json.
4. **Prediction Logic**
   - Encapsulated in the Energy Consumption class within utils.py.
5. **Flask Backend**
   - REST API routes implemented in app.py to handle requests and responses.
6. **User Interface**
   - Frontend built using **Bootstrap** and **JavaScript** (index.html) for interactive input and real-time predictions.

7. System Architecture

The system is modular and includes the following components:
- __init__.py → Initializes the project environment.
- config.py → Stores configuration paths (model, metadata, port settings).
- project_data.json → Stores categorical mappings and feature structures.
- utils.py → I mplements prediction logic through EnergyConsumption class.
- app.py → Hosts Flask API and routes for home and prediction.
- index.html → Web-based user interface for inputs and displaying results.
- Linear_Model.pkl → Pre-trained regression model file.
- energy_project_files.ipynb → Jupyter Notebook for EDA, preprocessing, and training.

8. Technology Stack

- Programming Language: Python
- Backend Framework: Flask
- Frontend: Bootstrap, HTML, CSS, JavaScript (Fetch API)
- Libraries: NumPy, Pandas, Scikit-learn, Joblib/Pickle, json
- Deployment: Cloud/server ready with host='0.0.0.0'

9. Workflow Diagram

The system workflow can be summarized as follows:

User Input (UI: index.html)
↓
Flask Backend (app.py)
↓
EnergyConsumption Class (utils.py)
↓
Model & Metadata (Linear_Model.pkl + project_data.json)
↓
Prediction Result (JSON Response)
↓
Displayed in Browser (Bootstrap UI)

10. Potential Clients

The Energy Consumption Predictor can provide significant value across multiple industries and sectors. Potential clients include:

1. **Real Estate Developers & Builders**
    - Estimate energy requirements for **new residential or commercial projects** before construction.
    - Improve **marketing and sales strategies** by providing buyers with estimated energy costs.
2. **Facility & Property Managers**
    - Monitor and optimize **energy usage in existing buildings**.
    - Identify opportunities for **retrofitting with energy-efficient systems**.
    - Reduce operational costs through **predictive maintenance and consumption planning.**
3. **Industrial Owners & Manufacturers**
    - Forecast energy needs for **production plants and factories**.
    - Optimize **shift planning and resource allocation** to reduce peak demand charges.
    - Integrate predictions with **IoT devices for smart factory operations.**
4. **Smart City Planners & Government Agencies**
    - Use predictions to design **sustainable infrastructure and urban energy grids**.
    - Support **policy-making** for renewable energy adoption and carbon reduction targets.
    - Enable **public-private partnerships** for green initiatives.
5. **Utility Providers & Energy Consultants**
    - Predict demand across customer segments for **better load balancing and grid stability**.
    - Offer **personalized energy-saving recommendations** to clients.
    - Improve forecasting accuracy for **pricing and supply management**.

11. Future Scope

To enhance the project further, the following areas are identified:
- Integration with real-time IoT sensor data for live monitoring.
- Incorporation of renewable energy optimization strategies.
- Implementation of advanced deep learning models (e.g., LSTM) for time-series forecasting.
- Development of a mobile application for wider accessibility.
- Integration with dynamic energy pricing APIs for cost predictions.

12. References

- Scikit-learn Documentation: https://scikit-learn.org/stable/
- Flask Documentation: https://flask.palletsprojects.com/
- Pandas Documentation: https://pandas.pydata.org/
- NumPy Documentation: https://numpy.org/
- Bootstrap Framework: https://getbootstrap.com/
- Research Paper: Energy Consumption Forecasting with Machine Learning Techniques (various sources)
- Open Source Tutorials & GitHub repositories on energy consumption prediction