ISSN (Online): 2455-3662
EPRA International Journal of Multidisciplinary Research (IJMR) - Peer Reviewed Journal
Volume: 12| Issue: 4| April 2026|| Journal DOI: 10.36713/epra2013 || SJIF Impact Factor 2026: 8.753 || ISI Value: 1.188

SMART ENERGY CONSUMPTION OPTIMIZATION USING MACHINE LEARNING

Sangeeth S R[1], Mrs. Vinitha.J[2]

[1]*UG Student, Department of Artificial Intelligence and Machine Learning, Dr. N.G.P. Arts and Science College, Coimbatore.*

[2]*Assistant professor, Department of Artificial intelligence and Machine Learning, Dr. N.G.P. Arts and Science College, Coimbatore.*

ABSTRACT

As modern living and industrial spaces grow more complex, the global demand for electricity is reaching levels we have never seen before. Traditional ways of managing energy – which usually involve manual checks or simply waiting for a monthly bill – are no longer effective in our fast-paced world. This research introduces a smarter way to manage power by using Machine Learning to turn raw electrical data into helpful, real-time insights. By using a Random Forest model trained on the Kaggle Household Electric Power Consumption dataset, our system achieved a predictive accuracy (R2 score) of 0.91. We built a modular solution featuring a smooth web dashboard, a Flask- powered bridge, and a high-speed prediction engine. The result is a tool that helps people see exactly where their energy goes and how to save it, providing answers in less than a second to support a more sustainable and budget-friendly future.

KEYWORDS: *Smart Energy, Optimization, Machine Learning, Sustainability, Flask API, Data Science, Predictive Analytics.*

1.INTRODUCTION

Today's world is changing rapidly, and as our homes and industries become more advanced, they are more power-hungry than ever before. This massive surge in electricity demand has exposed the flaws in our old, manual ways of monitoring energy. We can no longer afford to be reactive; we need a smarter, data-driven way to manage our resources.

Our project bridges the gap between complicated data and real-world decisions. In an era where every watt of power matters, we have designed a system that anticipates energy needs instead of just reporting them after the fact. By using intelligent algorithms, we transform electricity from an unmanaged utility into a strategic asset.

1.1 Overview of the Project

The heart of this project is a high-speed system that forecasts energy use with im- pressive accuracy. We designed it as a complete, end-to-end solution: from a ma- chine learning "brain" that does the heavy lifting to a user-friendly dashboard. By choosing a Random Forest model, we ensured the system can handle the messy, non-linear patterns of daily life, making energy management simple and accessible for everyone.

2.SYSTEM STUDY

Before building the system, we took a deep look at how energy is currently managed to understand exactly where the traditional methods fall short.

2.1 Existing System

Currently, energy management is mostly an afterthought. Most people use their appliances throughout the month and only realize how much they've consumed when the utility bill arrives. Traditional setups lack a centralized digital hub, leaving data trapped in old-fashioned meters that can't talk to modern web tools.

2.2 Proposed System

We propose a shift toward an intelligent, automated energy experience. Our system replaces manual guesswork with a Random Forest model that predicts usage patterns with high precision. To make this sophisticated technology useful, we built a web-based dashboard using HTML5 and JavaScript.

3.DESIGN AND ARCHITECTURE

Our design philosophy was to keep the technical complexity hidden and the user experience simple. We used a three-tier architecture that separates the data science from the visual dashboard.

The system works in two stages. First is the "Offline" stage, where we teach the model using Kaggle data. Second is the "Online" stage, where the Flask server loads that model and waits for users to ask questions.

4.SYSTEM REQUIREMENTS

The system needs a multi-core processor (like an Intel i5) and at least 8GB of RAM so the system can quickly clean and process large-scale datasets. Our entire environment is built on the Python 3.x ecosystem, using Flask, Pandas, and Scikit-learn.

5.SYSTEM WORKFLOW

We broke the system's operation into two logical phases to ensure speed and reliability.

5.1 Serving the User

When a user visits the dashboard, the process is lightning-fast:

ISSN (Online): 2455-3662
EPRA International Journal of Multidisciplinary Research (IJMR) - Peer Reviewed Journal
Volume: 12| Issue: 4| April 2026|| Journal DOI: 10.36713/epra2013 || SJIF Impact Factor 2026: 8.753 || ISI Value: 1.188

1. **Input**: User enters parameters (volt- age, global intensity, etc.).
2. **Calculation**: The server uses the pre- trained Joblib model to run a forecast.
3. **Check:** If the power prediction is > 5.0 kW, the system flags it as ”High Consumption.”
4. **Advice:** The dashboard shows the prediction and an energy-saving tip.

6. IMPLEMENTATION AND RE-SULTS

We brought this project to life using Scikit- learn. By using Joblib for model persistence, we ensured that users never have to wait more than a second for their forecast.

Table 1: System Performance Results

Metric	The Result
R^2 Score	0.91
MAE	0.22
Dataset Size	Over 200,000 habits
Speed	Under 1 second

The results show that our system is effective at spotting non-linear habits.

7.CONCLUSION

Our system successfully closes the gap be- tween complex data science and daily life. By combining a high-performance Random Forest model with a clean, friendly web dashboard, we've made energy optimization accessible to everyone.

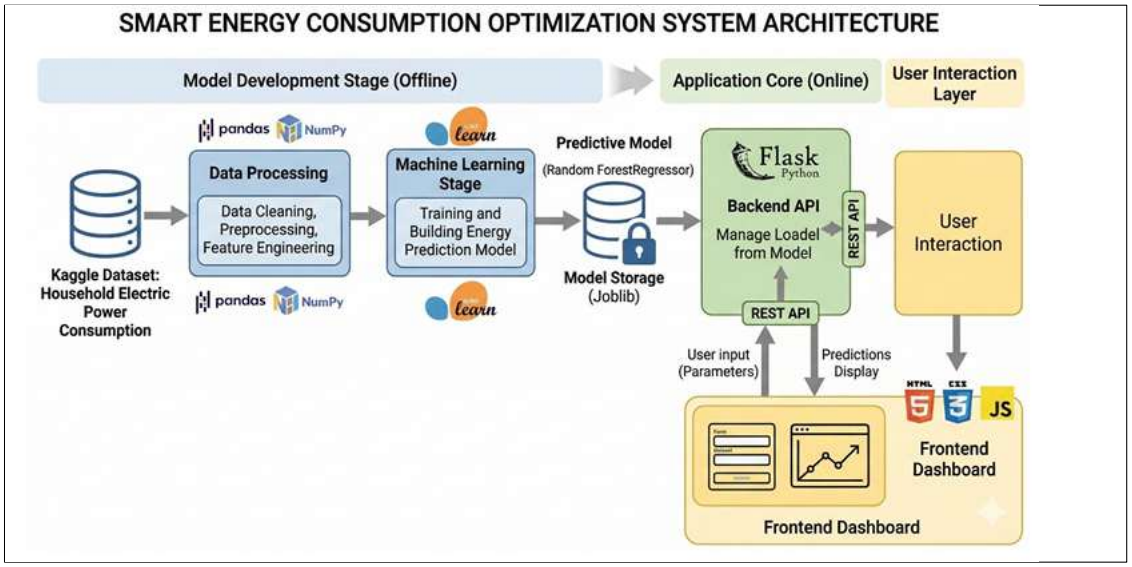


Figure 1: The modular three-tier architecture spanning the full page width for clarity.



Figure 2: A comprehensive step-by-step path from raw data input to user feedback.

ISSN (Online): 2455-3662
EPRA International Journal of Multidisciplinary Research (IJMR) - Peer Reviewed Journal
Volume: 12| Issue: 4| April 2026|| Journal DOI: 10.36713/epra2013 || SJIF Impact Factor 2026: 8.753 || ISI Value: 1.188

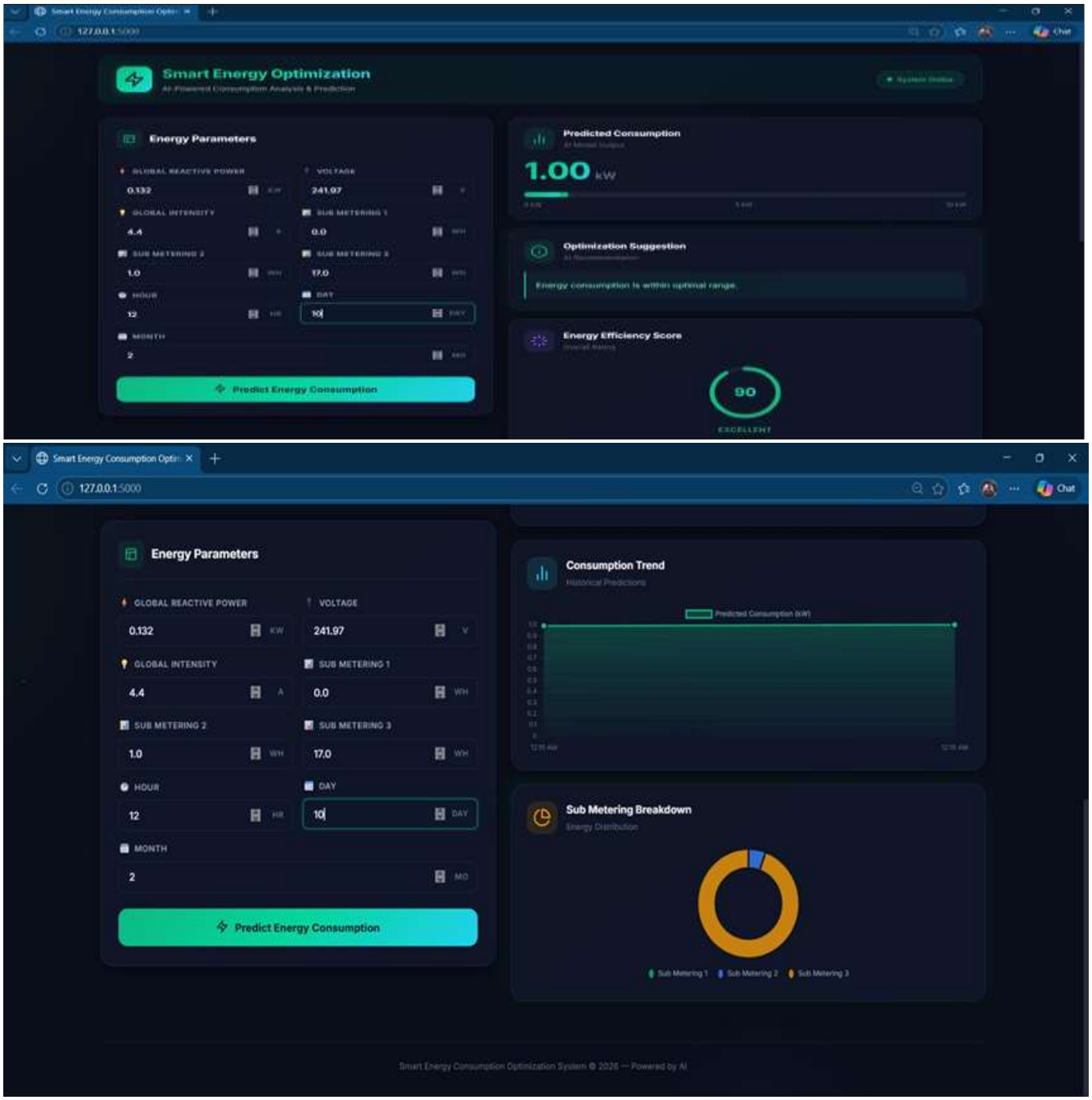


Figure 3: A detailed view of the final inter-face.

8.REFERENCES

1. *Scikit-learn Developers, "Random Forest Documentation," 2024.*
2. *Pallets Projects, "Flask Web Frame work Guide," 2023.*
3. *G. J. Hebrail, "Household Electric Power Consumption Dataset," Kaggle.*
4. *A. Géron, Hands-On Machine Learning with Scikit-Learn, 2022.*
5. *M. Grinberg, Flask Web Development, 2018.*