AI for Renewable Energy Management

Group Name: F A N U M
Gaganpreet Singh[202211023], Pragya Pranati[202211068], Suryansh Singh Raghuvansh[202211093]

***Abstract*—The increasing reliance on renewable energy sources, such as solar and wind, necessitates accurate forecasting to optimize grid integration and ensure energy stability. This research presents a predictive framework for solar and wind energy generation based on critical environmental and operational parameters. Leveraging advanced artificial intelligence (AI) techniques, we develop and implement separate, optimized models tailored to the unique characteristics of solar and wind energy prediction.**

For solar energy forecasting, factors such as solar irradiance, temperature, and cloud cover are analyzed, while wind energy prediction incorporates wind speed, direction, and atmospheric pressure. The models are evaluated on performance metrics such as accuracy, computational efficiency, and scalability. The results demonstrate that the selected AI models outperform conventional methods, providing reliable predictions with minimal error. This project not only contributes to improving renewable energy management but also highlights the importance of customized AI solutions for diverse energy domains.

The findings have significant implications for the energy sector, particularly in enhancing the efficiency of renewable energy systems, reducing reliance on fossil fuels, and supporting the transition to sustainable energy.

Keywords: AI algorithms, Renewable Energy, Prediction, Forecasting, Parameters, Solar Energy, Wind Energy

I. Introduction

INThe global transition towards renewable energy has underscored the critical need for accurate forecasting of solar and wind power generation. Solar and wind energy are inherently variable and dependent on fluctuating environmental factors, making their prediction a complex but essential task for energy system optimization. Accurate forecasting not only aids in balancing energy supply and demand but also supports grid stability and enhances the economic viability of renewable energy systems. In this research, we present a comprehensive approach to predicting solar and wind energy generation using advanced artificial intelligence (AI) models, carefully selected and optimized for their respective applications.

Our methodology begins with an in-depth exploratory analysis of potential models, tailored separately for solar and wind predictions. We systematically evaluated a variety of AI models, ranging from traditional machine learning algorithms to advanced deep learning techniques, to identify the most suitable models for each energy source. The selection process was guided by the unique characteristics of solar and wind energy generation, incorporating domain-specific factors such as solar irradiance, temperature, wind speed, and atmospheric pressure.

To rigorously assess model performance, we employed multiple evaluation metrics, including R-squared error, adjusted R-squared error, mean squared error (MSE), and mean absolute error (MAE). These metrics provided a comprehensive understanding of each model's predictive accuracy, consistency, and generalizability. The iterative analysis enabled us to identify models that achieved optimal performance for solar and wind predictions individually.

Once the best-performing models were identified, we transitioned from the development phase to deployment. The models were operationalized through web-based portals, designed for accessibility and usability. These portals were developed using Flask, a lightweight yet powerful web application framework, and hosted on a dedicated server to ensure seamless access for end users.

Thus, our methodology boils down to a structured two-step plan that ensures both analytical rigor and practical usability:

Analysis and Model Selection:

- Conducted an in-depth analysis of various AI models tailored for solar and wind energy prediction.
- Evaluated models based on environmental and operational parameters such as solar irradiance, temperature, wind speed, and atmospheric pressure.
- Employed multiple evaluation metrics, including R-squared error, adjusted R-squared error, mean squared error (MSE), and mean absolute error (MAE), to rigorously assess model performance.
- Identified the most accurate and efficient models for solar and wind predictions individually.

Deployment and Accessibility:

- Operationalized the selected models into accessible, user-friendly web portals.
- Utilized Flask, a lightweight web application framework, for server hosting to ensure seamless interaction with the deployed models.
- Demonstrated the scalability and practical usability of the framework, enabling real-time forecasting for stakeholders.

This two-step approach bridges the gap between theoretical model development and practical implementation. The analysis phase ensures the adoption of optimal models for each energy source, while the deployment phase translates these findings into scalable, real-world solutions. By integrating advanced AI techniques with an accessible user interface, this research contributes significantly to renewable energy forecasting, supporting sustainable energy transitions and fostering greater adoption of renewable technologies.

II. Analysis of Dataset and Models

A. Objective and Dataset Overview

The primary objective of the analysis was to systematically evaluate multiple regression models to determine the most accurate and reliable predictors for solar and wind energy generation. The analysis utilized a dataset with time-indexed environmental features and energy generation outputs. Key features included:

- **Solar Energy:** Solar irradiance (SWTDN, SWGDN), temperature (T), and actual solar energy generation (DE_solar_generation_actual).
- **Wind Energy:** Wind speeds at different levels (v_1, v_2, v_50m), surface roughness (z_0), and actual wind energy generation (DE_wind_generation_actual).
- **Dataset Split:** The dataset was split into 70% for training and 30% for testing, ensuring adequate data for both model fitting and evaluation.

B. Model Selection Process

To streamline model selection, the LazyPredict library was employed. LazyPredict evaluates a wide range of regression models and outputs performance metrics, including R-squared, Mean Squared Error (MSE), and Mean Absolute Error (MAE).

- **Solar Energy Prediction:**
  - LazyPredict applied to solar-related features identified HistGradientBoostingRegressor as the best-performing model, with high R-squared values and low error metrics.
  - Visualizations of predicted vs. actual values revealed strong alignment, validating the model's capability to capture complex relationships.

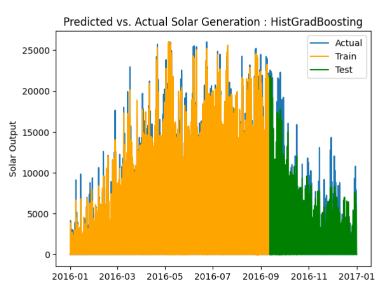

Figure 1: Predicted vs. Actual Solar Generation : HistGradBoosting

- **Wind Energy Prediction:**
  - For the wind dataset, GradientBoostingRegressor emerged as the top model, excelling in both accuracy and consistency during LazyPredict evaluation.
  - Similar to solar, the evaluation included R-squared, MSE, and MAE metrics along with strong alignment with the model's capability to capture complex relationships which was evident by the Visualizations of predicted vs. actual values.

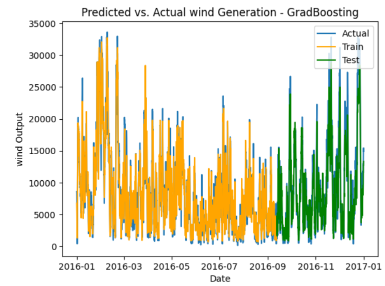

Figure 2: Predicted vs. Actual Wind Generation - GradBoostingRegressor

C. Model Validation

To ensure robustness, the best models identified for solar and wind energy were further evaluated using a 5-fold cross-validation approach. This provided an average performance score across different data subsets, minimizing risks of over-fitting or data dependency.

- **Solar Model:** HistGradientBoostingRegressor achieved consistent R-squared values across folds, with minimal deviation.

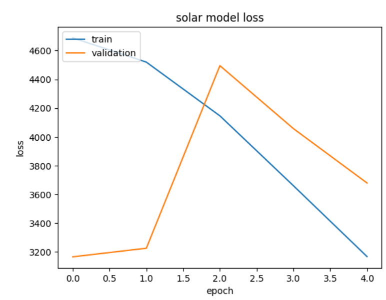

Figure 3: Solar Model Loss - First Draft

- **Dataset Model:** GradientBoostingRegressor similarly exhibited reliable performance across folds.

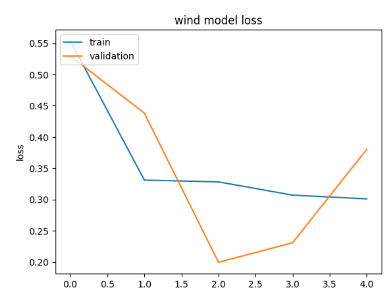

Figure 4: Wind Model Loss - First Draft

D. Results and Insights

- **Model Performance:**
  - The HistGradientBoostingRegressor outperformed other regressors for solar predictions due to its ability to handle non-linear relationships and heteroscedasticity in the data.

  - For wind predictions, GradientBoostingRegressor excelled, highlighting its capacity for managing multi-dimensional feature interactions.
- **Error Metrics:**
  - Error values (MSE, MAE) for both models were significantly lower compared to other regressors, emphasizing their suitability for deployment.
- **Predicted vs. Actual Plots:**
  - Clear alignment between predicted and actual values for both training and testing sets validated the models' generalization ability.

E. Conclusion

This analysis identified the best regression models for solar and wind energy prediction using an automated, systematic approach. The LazyPredict library allowed efficient comparison of multiple regressors, while cross-validation confirmed the robustness of the chosen models. The results highlight the readiness of these models for deployment in forecasting systems, with implications for enhancing renewable energy management and integration.

III. Solar Energy Prediction Analysis

A. Objective and Dataset Overview

The script aims to develop and deploy a machine learning model for solar energy prediction. It utilizes historical and environmental data to train a predictive model and save it for deployment.

- **Feature Selection:** The script loads solar radiation (SWTDN, SWGDN), temperature (T) as input and actual solar energy generation (DE_solar_generation_actual) as output.
- **Dataset Split:** The dataset was split into 70% for training and 30% for testing, ensuring adequate data for both model fitting and evaluation.

B. Model Training

- **Algorithm Used:**
  - **HistGradientBoostingRegressor:** A high-performance gradient boosting algorithm suitable for tabular data.
- **Training Process:**
  - The model is trained on the selected features from the training dataset.
  - The script ensures reproducibility and consistent performance by using standard scikit-learn practices.

C. Model Deployment

- **Serialization:** The trained model is serialized and saved as a .pkl file using Python's pickle module, ensuring it can be reused without retraining.
- **Verification:** The script reloads the model from the saved file and validates its functionality by making a sample prediction on synthetic input data.

D. Deployment Feasibility

The use of serialized .pkl files makes the model ready for integration into applications, such as web portals or energy management systems. This facilitates:

- **Scalability:** Easy distribution across platforms.
- **Real-time prediction:** Fast inference capabilities on new data.

E. Results and Insights

- **Model Choice:** The HistGradientBoostingRegressor was selected for its robust handling of non-linear relationships and strong performance in prior evaluations.
- **Serialization:** The model was successfully serialized and verified, ensuring practical usability for deployment.
- **Prediction Example:** The model correctly performed predictions on test inputs, confirming its readiness for real-world applications.

F. Conclusion

This analysis demonstrates a systematic approach to solar energy prediction, from data preparation and model training to deployment readiness. The choice of HistGradientBoostingRegressor, combined with efficient serialization, ensures both accuracy and usability. This framework is adaptable for broader renewable energy forecasting tasks, with potential scalability to other regions and datasets.

IV. Wind Energy Prediction Analysis

A. Objective and Dataset Overview

The script aims to develop and deploy a machine learning model for predicting wind energy generation based on environmental and operational parameters.

- **Feature Selection:** The script loads wind speed at various heights (v_1, v_2, v_50m) and surface roughness (z_0) as input and actual wind energy generation (DE_wind_generation_actual) as output.
- **Dataset Split:** The dataset was split into 70% for training and 30% for testing, ensuring adequate data for both model fitting and evaluation.

B. Model Training

- **Algorithm Used:**
  - **GradientBoostingRegressor:** A robust gradient boosting algorithm that excels in modeling non-linear relationships.
- **Training Process:**
  - The model is trained using the selected wind-related features.
  - It optimizes the prediction of DE_wind_generation_actual through iterative improvements.

C. Model Deployment

- **Serialization:** The trained model is saved as a .pkl file using the pickle module. This ensures the model is easily reusable without requiring retraining.
- **Verification:** The script reloads the model from the saved file and validates its functionality by making a sample prediction on synthetic input data.

D. Deployment Feasibility

Serialization into .pkl files prepares the model for integration into various platforms, such as:

- **Energy Management Systems:** For grid-level optimization operations and systems.
- **Web Applications:** Enabling real-time wind energy prediction which is easily accessible and usable.

E. Results and Insights

- **Model Choice:** GradientBoostingRegressor was selected due to its demonstrated accuracy and adaptability to complex feature interactions in prior analyses.
- **Serialization:** The model was successfully serialized and verified, ensuring practical usability for deployment.
- **Prediction Example:** The model provided accurate predictions for sample inputs, validating its practical utility.

F. Conclusion

This analysis presents a systematic approach to wind energy prediction, covering data preparation, model training, and deployment readiness. The choice of GradientBoostingRegressor, combined with efficient serialization, ensures the model's scalability and usability in real-world applications. This methodology lays a foundation for future enhancements, such as integrating dynamic input sources for real-time predictions.

V. Analysis of Flask-Based System for Model Deployment

A. System Overview

The project consists of two independent but structurally similar Flask-based applications for predicting solar and wind energy generation. Each application employs a pre-trained machine learning model developed during the project. The machine learning models were trained on domain-specific data, such as weather parameters (e.g., temperature, solar irradiance, wind speed), to predict energy generation in kilowatts. The deployed systems coordinate model inference with user interaction, enabling dynamic and real-time prediction capabilities.

B. User Interaction and Deployment Coordination

The deployment aligns the project's core prediction models with an interactive user interface to bridge the gap between AI and end-users:

- **HTML Templates and User Forms:**
  - The home page (/) serves as the primary user interface, where users can input feature values such as solar irradiance, temperature, or wind speed. Templates like solar_prediction.html and wind_prediction.html provide a structured form layout.
  - Flask renders the prediction results back to the user, enhancing interactivity.
- **Dynamic and Real-Time Predictions:**
  - Once user inputs are submitted, the server processes the data and returns results dynamically, offering near-instant feedback. This capability enables stakeholders, such as energy managers or researchers, to evaluate scenarios in real-time.

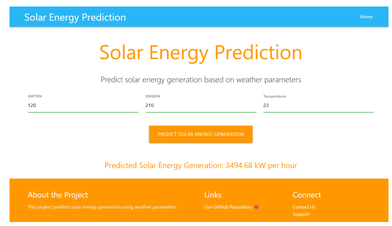

Figure 5: Solar Energy Prediction Web Portal

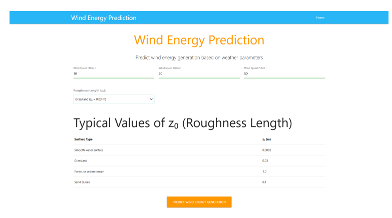

Figure 6: Wind Energy Prediction Web Portal

C. Model-Specific Deployment Integration

- **Solar Energy Prediction:**
  - The solar model is trained using features that include solar irradiance, temperature, and other meteorological data. The Flask app reads these inputs, processes them into a vector, and passes them to the model.
  - After inference, the app displays the predicted solar energy generation (in kW) directly on the webpage.
  - These predictions can be used by utility operators can plan photovoltaic (PV) system outputs based on predicted energy availability.
- **Wind Energy Prediction:**
  - The wind prediction model relies on features such as wind speed, air pressure, and air density to estimate energy generation. These features are processed similarly to the solar application.
  - The app enables real-time predictions, aiding in wind farm operational planning and power output estimation.

  Both applications emphasize seamless communication between the trained AI models and the user, ensuring the underlying algorithms remain abstracted from the interface.

D. Advantages and Applications

- **Scalable Deployment for Real-World Impact:** The project's design, enabling the easy deployment of updated models or integration of new energy prediction features, provides the flexibility needed to adapt to evolving renewable energy scenarios.
- **Actionable Insights for Energy Management:** Real-time predictions aid decision-making for grid operations, energy storage planning, and load balancing in solar and wind power systems.
- **Improved Accessibility:** The user-facing interface democratizes access to AI predictions, making them usable by non-technical stakeholders in energy planning.

E. Conclusion

This project effectively demonstrates how AI models for solar and wind energy prediction can be operationalized using Flask. By coordinating model development with a user-facing deployment platform, the system bridges theoretical innovation and practical utility. This approach can be extended to other renewable energy forecasting tasks, contributing to more efficient and sustainable energy management systems.

VI. References

1) Artificial intelligence applications for microgrids integration and management of hybrid renewable energy sources[Link].
2) IET Renewable Power Gen - 2022 - Applications of artificial intelligence in renewable energy systems [Link].
3) Present and Future of AI in Renewable Energy[Link].

VII. GitHub Repository

The GitHub Repository to the project can be accessed here