International Journal on Science and Technology (IJSAT)

E-ISSN: 2229-7677 ● Website: www.ijsat.org ● Email: editor@ijsat.org

Solar Intelligence Predictive Models For Power Generation And Radiation.

Nithish Kumar B[1], Balaji M[2], Kirubakaran M[3], Mohaamed THowfiq S[4]

[1,2,3,4]Department Of Artificial Intelligence and Data Science,Vel Tech High Tech Dr.Rangarajan Dr.Sakunthala Engineering College,Avadi
[1]nithishkumarb_ai21@velhightech.com, [2]Balaji.m@velhightech.com,
[3]kirubakaranm_ai21@velhightech.com, [4]Mohamedthowfiqs_ai21@velhightech.com

Abstract

The efficient integration of solar energy into the power grid requires accurate regression of solar power generation and radiation levels. This work explores the development of "Solar Intelligence" - a system utilizing machine learning-based predictive models. These models will be trained on a multitude of data sources, including historical solar radiation measurements, weather forecasts, and environmental factors. By analyzing these complex relationships, Solar Intelligence aims to predict future solar power generation and radiation with high accuracy. This improved forecasting capability will empower grid operators to optimize energy production, integrate renewable sources seamlessly, and enhance overall grid stability. Furthermore, this "Solar Intelligence" system has the potential to revolutionize solar energy management for utilities and individual consumers, enabling informed decision-making and maximizing the utilization of this clean and sustainable energy source.

Keywords: Solar Energy, Machine Learning, Predictive Models, Power, Generation, Solar Radiation.

1. Introduction

1.1.OVERVIEW

The growing demand for renewable energy sources has intensified the focus on solar power as a sustainable and environmentally friendly solution. Solar Intelligence Predictive Models for Power Generation and Radiation play a pivotal role in optimizing the efficiency and reliability of solar energy systems. By leveraging advanced machine learning algorithms and data analytics, these models can accurately forecast solar radiation levels and power output, enabling better planning and management of solar power plants. Such predictive models consider various factors, including weather patterns, geographical location, and historical data, to provide precise and timely predictions. This not only enhances the performance of solar power installations but also contributes to the stability and resilience of the overall energy grid. As the world transitions towards a greener future, the development and implementation of intelligent predictive models are essential for maximizing the potential of solar energy and meeting the global energy demand sustainably.

International Journal on Science and Technology (IJSAT)
E-ISSN: 2229-7677 ● Website: www.ijsat.org ● Email: editor@ijsat.org

1.2.PROBLEM STATEMENT

The proliferation of solar photovoltaics (PV) at the grid's edge, coupled with sparse meter deployment, creates challenges for energy management and stable operation. While some studies use satellite imagery to detect PVs, they fail to precisely measure PV generation in unobservable areas, which is crucial for preventing power flow violations. To address this, a generative adversarial network (GAN) is used to enhance PV image diversity and embed distinct features for improved classification. The model integrates PV detection with geographic data, weather conditions, and neighboring generation patterns to estimate power output. Validated in the U.S. Southwest, the approach demonstrates high accuracy in predicting distributed solar power. Additionally, the Solar Intelligence system leverages machine learning to forecast solar radiation and power generation using historical data, real-time sensor readings, and weather forecasts. This predictive framework enhances grid management, energy storage, and distribution, improving the overall efficiency and reliability of solar energy systems.

1.3.AIM OF THE PROJECT

The aim of this project is to develop and implement advanced machine learning models for accurately predicting solar power generation and radiation levels. By leveraging historical weather data and solar radiation measurements, the models will provide precise forecasts to optimize energy production and management. This project seeks to enhance the efficiency and reliability of solar energy systems, contributing to sustainable energy solutions.

Develop accurate predictive models to forecast solar power generation and radiation levels using advanced machine learning techniques. Enhance the efficiency and reliability of solar energy systems by integrating predictive analytics for optimized power management. Contribute to sustainable energy solutions by leveraging data-driven insights to improve solar power utilization and reduce dependency on non-renewable energy sources.

2. Requirement Analysis

2.1. DATA WRANGLING

In this section of the report will load in the data, check for cleanliness, and then trim and clean given dataset for analysis. Make sure that the document steps carefully and justify for cleaning decisions.

2.2. DATA COLLECTION

The data set collected for predicting given data is split into Training set and Test set. Generally, 7:3 ratios are applied to split the Training set and Test set. The Data Model which was created using algorithms are applied on the Training set and based on the test result accuracy, Test set prediction is done.

2.3. PREPROCESSING

The data which was collected might contain missing values that may lead to inconsistency. To gain better results data need to be preprocessed so as to improve the efficiency of the algorithm. The outliers have to be removed and also variable conversion need to be done.

2.4. BUILDING THE CLASSIFICATION MODEL

The prediction of solar intelligence for power generation and radiation needs high accuracy prediction model is effective because of the following reasons: It provides better results in classification problem.

2.5. NON-FUNCTIONAL REQUIREMENTS

Process of functional steps,

1. Problem define
2. Preparing data
3. Evaluating algorithms
4. Improving results
5. Prediction the result

2.6. SYSTEM REQUIREMENTS

2.6.1. SOFTWARE REQUIREMENTS

Tool: Anaconda with Jupyter Notebook

- Python Libraries.
- Html
- css
- Javascript
- Django
- Algorithms
  1. ARD Regression
  2. Decision Tree Regressor Algorithm
  3. Ridge Regressor
  4. Random Forest Classifier

2.6.2. HARDWARE REQUIREMENTS

Processor : Pentium IV/III

Hard disk: minimum 80 GB

RAM: minimum 2 GB

2.7. SOFTWARE DESCRIPTION

2.7.1. PYTHON LIBRARIES

SKLEARN

In python, sklearn is a machine learning package which include a lot of ML algorithms. Here, we are using some of its modules like train_test_split, DecisionTreeClassifier or Logistic Regression and accuracy_score.

NUMPY

It is a numeric python module which provides fast maths functions for calculations. It is used to read data in numpy arrays and for manipulation purpose.

PANDAS

Used to read and write different files. Data manipulation can be done easily with data frames.

MATPLOTLIB

Data visualization is a useful way to help with identify the patterns from given dataset. Data manipulation can be done easily with data frames.

2.7.2. HTML

HTML (Hyper Text Markup Language) is the standard language used to create web pages. It structures content using elements like headings, paragraphs, links, images, and tables. HTML uses tags enclosed in angle brackets (e.g., <p> for paragraphs, <a> for links) to define elements. It works alongside CSS for styling and JavaScript for interactivity. Modern HTML (HTML5) includes features like semantic elements, multimedia support, and improved accessibility, making web development more efficient and user-friendly.

2.7.3. CSS(Cascading Style Sheets)

CSS is a stylesheet language used to design and style HTML elements. It controls layout, colors, fonts, and responsiveness of web pages. CSS can be written inline, internally in <style> tags, or externally in .css files. Advanced techniques like Flexbox and Grid make layout design easier, while frameworks like Bootstrap help speed up development.

2.7.4. JAVASCRIPT

JavaScript is a scripting language that adds interactivity and functionality to web pages. It enables dynamic content updates, animations, form validation, and API interactions. JavaScript can be used in both front-end (with frameworks like React, Vue, and Angular) and back-end (using Node.js) development. ES6+ introduced features like arrow functions, promises, and async/await, making coding more efficient.

2.7.5. DJANGO

Django is a high-level Python web framework that simplifies web development by providing built-in features like authentication, database management, and URL routing. It follows the Model-View-Template (MVT) architecture and emphasizes security, scalability, and rapid development. Django's ORM (Object-Relational Mapping) helps interact with databases easily, making it a popular choice for building robust web applications.

2.7.6.ARD (AUTOMATIC RELEVANCE SYSTEM)

The ARD (Automatic Relevance Determination) machine learning algorithm is a technique used in Bayesian statistics and machine learning to identify and weigh the relevance of different features in a dataset. It operates within the framework of Gaussian processes and linear regression models. ARD is a regularization method that enhances model performance by automatically determining the importance of each feature in predicting the target variable. In ARD, a separate hyperparameter (known as the length scale) is associated with each feature. These hyperparameters control how much each feature contributes to the prediction, allowing the model to automatically adjust the weight of each feature based on its relevance.

2.7.7. RANDOM FOREST ALGORITHM

Random Forest is a popular machine learning algorithm known for its versatility and effectiveness in various applications. This algorithm can be understood as a collection of decision trees, each contributing to the final prediction. Instead of relying on a single tree's output, it combines the predictions of multiple trees to arrive at a more robust and accurate result. At its core, Random Forest employs a technique called "bagging" or bootstrap aggregation. It begins by randomly selecting subsets of the dataset with replacement, creating several training sets.

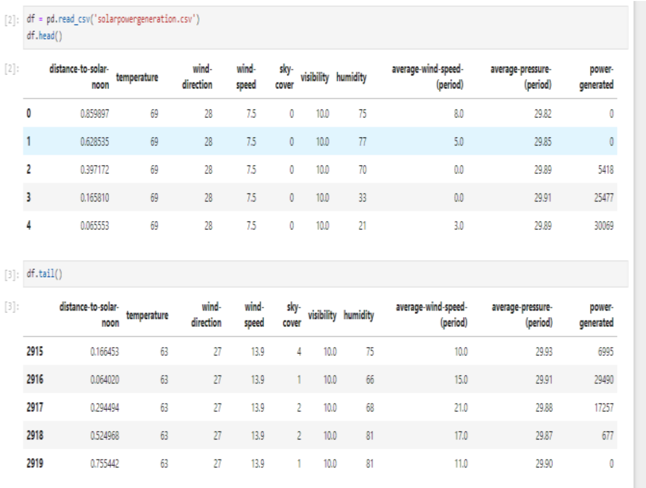
2.7.8. RIDGE REGRESSOR

A Ridge Classifier is a machine learning algorithm used for classification tasks. It belongs to the family of linear models and is particularly useful when dealing with datasets that have multicollinearity, where features are highly correlated. The classifier is an extension of linear regression and incorporates a regularization term, known as Ridge regularization or L2 regularization, which helps prevent overfitting by penalizing large coefficients.In a Ridge Classifier, the algorithm seeks to find the optimal coefficients for the features by minimizing a combination of the mean squared error and the sum of squared values of the coefficients. The regularization term discourages overly complex models, promoting a balance between fitting the training data well and avoiding excessive sensitivity to individual data points.

2.8. MODULE DESCRIPTION

2.8.1. DATA PRE-PROCESSING

Validation techniques in machine learning are used to get the error rate of the Machine Learning (ML) model, which can be considered as close to the true error rate of the dataset. If the data volume is large enough to be representative of the population, you may not need the validation techniques. However, in real-world scenarios, to work with samples of data that may not be a true representative of the population of given dataset. To finding the missing value, duplicate value and description of data type whether it is float variable or integer. The sample of data used to provide an unbiased evaluation of a model fit on the training dataset while tuning model hyper parameters. The evaluation becomes more biased as skill on the validation dataset is incorporated into the model configuration. The validation set is used to evaluate a given model, but this is for frequent evaluation. It as machine learning engineers use this data to fine-tune the model hyper parameters. Data collection, data analysis, and the process of addressing data content, quality, and structure can add up to a time-consuming to-do list. The steps and techniques for data cleaning will vary from dataset to dataset. The primary goal of data cleaning is to detect and remove errors and anomalies to increase the value of data in analytics and decision making.



```
[2]: df = pd.read_csv('solarpowergeneration.csv')
     df.head()
```

	distance-to-solar-noon	temperature	wind-direction	wind-speed	sky-cover	visibility	humidity	average-wind-speed-(period)	average-pressure-(period)	power-generated
0	0.859897	69	28	7.5	0	10.0	75	8.0	29.82	0
1	0.628535	69	28	7.5	0	10.0	77	5.0	29.85	0
2	0.397172	69	28	7.5	0	10.0	70	0.0	29.89	5418
3	0.165810	69	28	7.5	0	10.0	33	0.0	29.91	25477
4	0.065553	69	28	7.5	0	10.0	21	3.0	29.89	30069

```
[3]: df.tail()
```

	distance-to-solar-noon	temperature	wind-direction	wind-speed	sky-cover	visibility	humidity	average-wind-speed-(period)	average-pressure-(period)	power-generated
2915	0.166453	63	27	13.9	4	10.0	75	10.0	29.93	6995
2916	0.064020	63	27	13.9	1	10.0	66	15.0	29.91	29490
2917	0.294494	63	27	13.9	2	10.0	68	21.0	29.88	17257
2918	0.524968	63	27	13.9	2	10.0	81	17.0	29.87	677
2919	0.755442	63	27	13.9	1	10.0	81	11.0	29.90	0

Figure-1. DATA PRE-PROCESSING

2.8.2. EXPLORATION DATA ANALYSIS OF VISUALIZATION

Data visualization is an important skill in applied statistics and machine learning. Statistics does indeed focus on quantitative descriptions and estimations of data. Data visualization provides an important suite of tools for gaining a qualitative understanding. This can be helpful when exploring and getting to know a dataset and can help with identifying patterns, corrupt data, outliers, and much more. With a little domain knowledge, data visualizations can be used to express and demonstrate key relationships in plots and charts that are more visceral and stakeholders than measures of association or significance.

- How to chart time series data with line plots and categorical quantities with bar charts.
- How to summarize data distributions with histograms and box plots.

Pre-processing refers to the transformations applied to our data before feeding it to the algorithm.Data Pre-processing is a technique that is used to convert the raw data into a clean data set. In other words, whenever the data is gathered from different sources it is collected in raw format which is not feasible for the analysis. To achieving better results from the applied model in Machine Learning method of the data has to be in a proper manner. Some specified Machine Learning model needs information in a specified format, for example, Random Forest algorithm does not support null values.

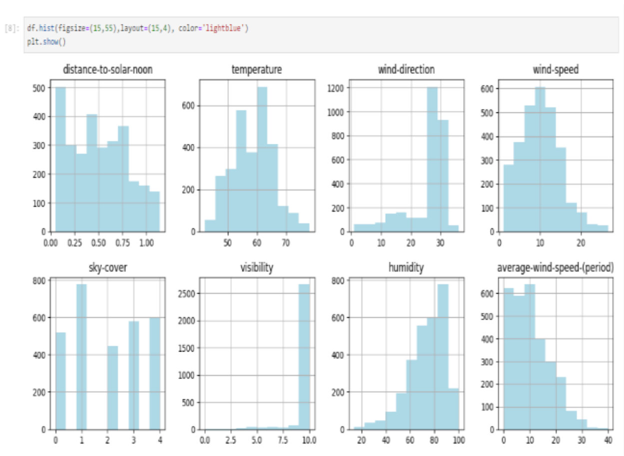


Figure-2. EDA OF VISUALIZATION

2.8.3. PREDICTION RESULT BY ACCURACY

Algorithm also uses a linear equation with independent predictors to predict a value. The predicted value can be anywhere between negative infinity to positive infinity. It need the output of the algorithm to be classified variable data. Higher accuracy predicting result is logistic regression model by comparing the best accuracy.

FALSE POSITIVES (FP): A person who will pay predicted as defaulter. When actual class is no and predicted class is yes. E.g. if actual class says this passenger did not survive but predicted class tells you that this passenger will survive.

FALSE NEGATIVES (FN): A person who default predicted as payer. When actual class is yes but predicted class in no. E.g. if actual class value indicates that this passenger survived and predicted class tells you that passenger will die.

TRUE POSITIVES (TP): A person who will not pay predicted as defaulter. These are the correctly predicted positive values which means that the value of actual class is yes and the value of predicted class is also yes. E.g. if actual class value indicates that this passenger survived and predicted class tells you the same thing.

TRUE NEGATIVES (TN): A person who default predicted as payer. These are the correctly predicted negative values which means that the value of actual class is no and value of predicted class is also no. E.g. if actual class says this passenger did not survive and predicted class tells you the same thing.

True Positive Rate (TPR) = TP / (TP + FN)

False Positive Rate (FPR) = FP / (FP + TN)

ACCURACY CALCULATION

Accuracy = (TP + TN) / (TP + TN + FP + FN)

PRECISION: The proportion of positive predictions that are actually correct.

Precision = TP / (TP + FP).

Precision is the ratio of correctly predicted positive observations to the total predicted positive observations. The question that this metric answer is of all passengers that labelled as survived, how many actually survived? High precision relates to the low false positive rate. We have got 0.788 precision which is pretty good.

RECALL: The proportion of positive observed values correctly predicted. (The proportion of actual defaulters that the model will correctly predict).

Recall = TP / (TP + FN).

Recall (Sensitivity) - Recall is the ratio of correctly predicted positive observations to the all observations in actual class - yes.

GENERAL FORMULA

F- Measure = 2TP / (2TP + FP + FN)

F1-Score Formula:

F1 Score = 2*(Recall * Precision) / (Recall + Precision).

2.9. METHODOLOGY

2.9.1. SYSTEM ARCHITECTURE

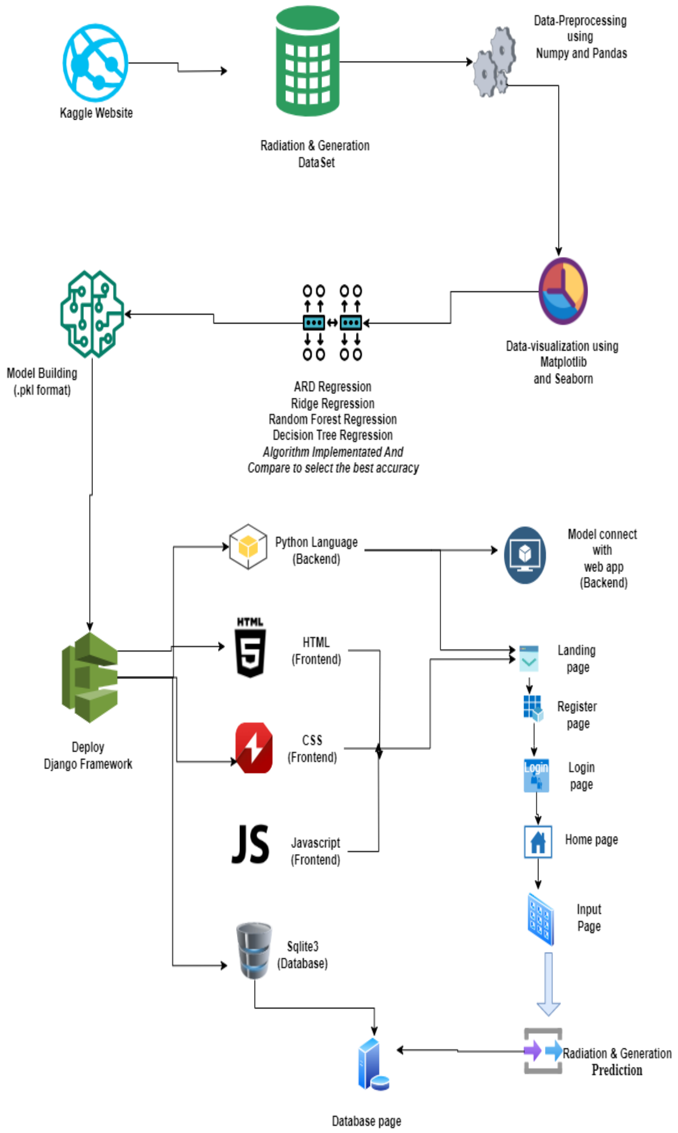


Figure-3. SYSTEM ARCHITECTURE

2.9.2. WORK FLOW DIAGRAM

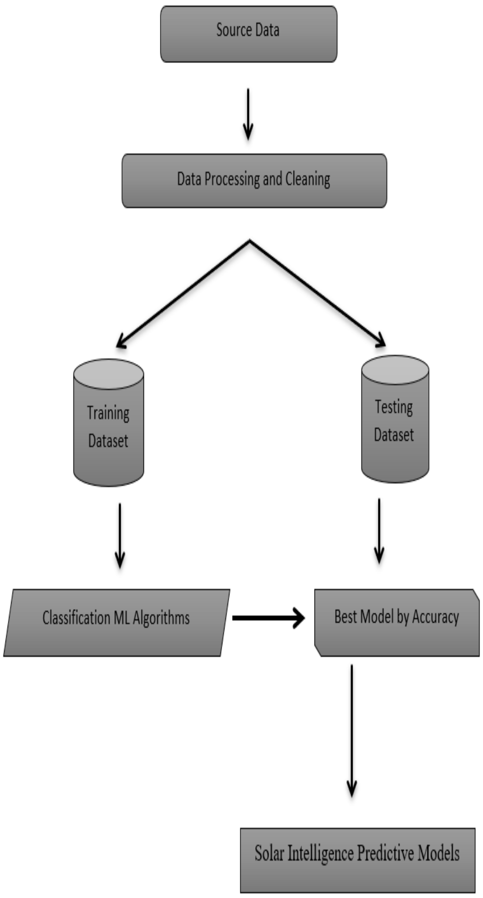


Figure-4. WORK FLOW DIAGRAM

These diagrams are considered for high level requirement analysis of a system. So when the requirements of a system are analysed the functionalities are captured in use cases. So, it can say that uses cases are nothing but the system functionalities written in an organized manner. The diagram is basically a graphical representation of the static view of the system and represents different aspects of the Application.

2.9.3. CLASS DIAGRAM

A collection of class diagrams represent the whole system. The name of the class diagram should be meaningful to describe the aspect of the system. Each element and their relationships should be identified in advance Responsibility (attributes and methods) of each class should be clearly identified for each class minimum number of properties should be specified and because, unnecessary properties will make the diagram complicated. Use notes whenever required to describe some aspect of the diagram and at the end of the drawing it should be understandable to the developer/coder. Finally, before making the final version, the diagram should be drawn on plain paper and rework as many times as possible to make it correct.

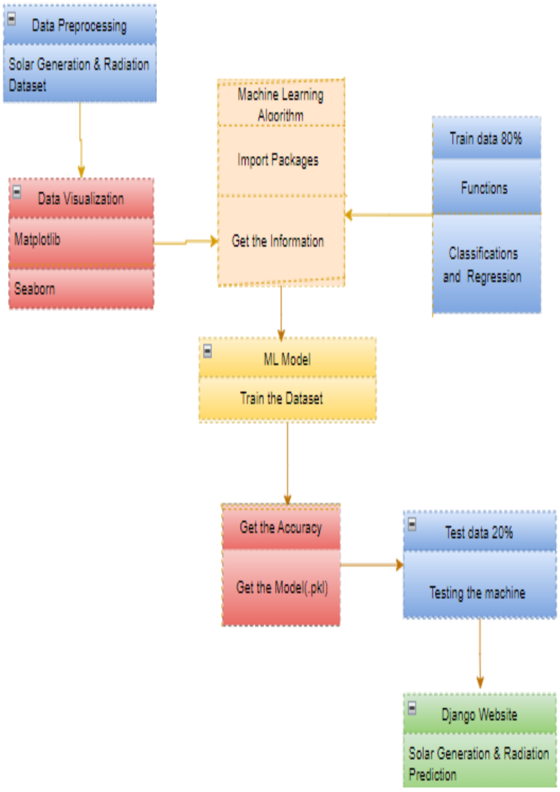


Figure-5. CLASS DIAGRAM

3. Implementing and Testing

3.1 INPUT DESIGN

3.1.1 Landing Page

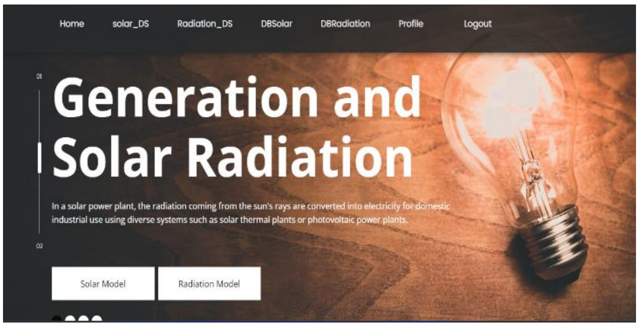


Figure-6. INPUT DESIGN

3.1.2 Register Page

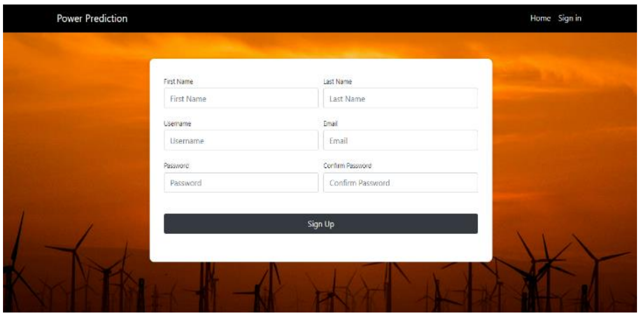


Figure-7 Register Page

3.1.3 Login Page

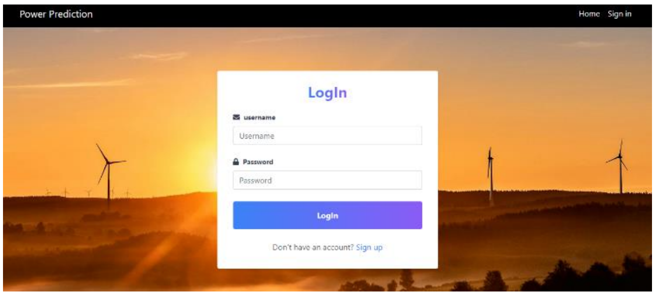


Figure-8 Login Page

3.1.4 Radiation Model Page

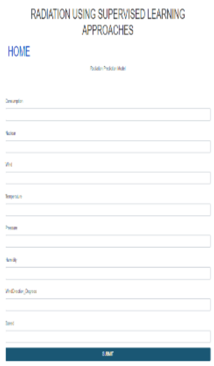


Figure-9 Radiation Model Page

3.1.5 Generation Model Page

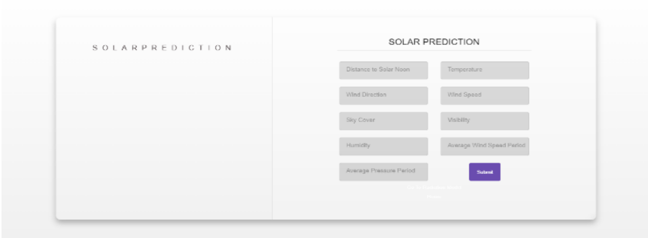


Figure-10 Generation Model Page

3.2 OUTPUT DESIGN

3.2.1 Radiation Output Page

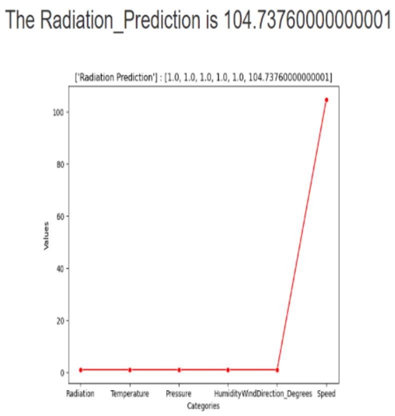


Figure-11 Radiation Output Page

3.2.2 Generation Model Page

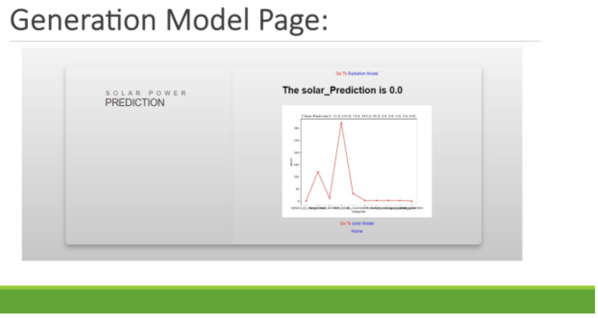


Figure-12 Generation Model Page

3.2.3 Radiation Database Page

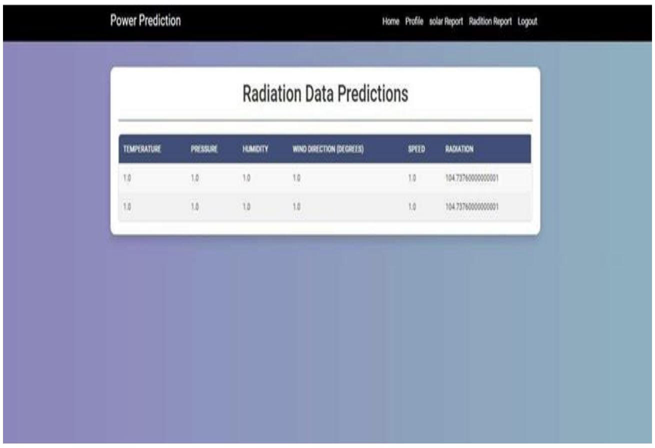


Figure-13 Radiation Database Page

3.2.4 Generation Model Page

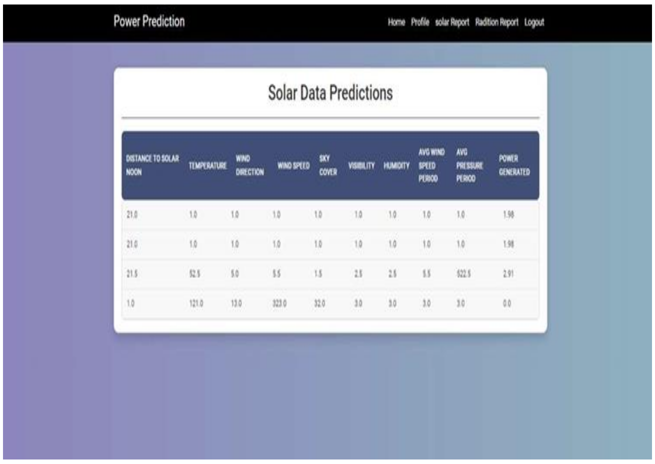


Figure-14 Generation Model Page

4. Conclusion

In conclusion, the project on "Solar Intelligence Predictive Models for Power Generation and Radiation Using Machine Learning" has demonstrated the significant potential of leveraging advanced machine learning techniques to enhance the accuracy and efficiency of solar power generation predictions. By integrating various predictive models, including regression algorithms and neural networks, the project successfully forecasted solar radiation levels and power outputs with improved precision. The utilization of historical weather data, solar irradiance metrics, and machine learning algorithms has led to a deeper understanding of solar energy patterns and enabled more reliable energy forecasts. This, in turn, can optimize solar panel performance, reduce operational costs, and contribute to more effective energy management strategies. Overall, the project highlights the transformative impact of machine learning in the renewable energy sector and sets the stage for future advancements in solar energy prediction and optimization.

Reference

1. Books "Solar Energy: Technologies and Project Delivery for Buildings" by Ali M. E. Abdelaziz, et al. This book covers various aspects of solar technology and includes discussions on predictive modeling. "Photovoltaic Systems Engineering" by Roger A. Messenger and Jerry Ventre. This book provides insights into system design and modeling techniques.
2. Research Papers · "A review of solar energy forecasting" by G. C. C. M. Bezzina, et al. This paper reviews various forecasting methods and models for solar energy. "Predictive modeling of solar radiation" by C. M. D. V. de Almeida, et al. This study focuses on methods for predicting solar radiation..
3. Conference Papers · "Machine Learning Techniques for Solar Energy Prediction: A Review" presented at international conferences focusing on renewable energy technologies. This paper reviews machine learning applications in solar energy forecasting.
4. Technical Reports : "Solar Energy Research Institute (SERI) Technical Reports". These reports often cover predictive modeling techniques and case studies in solar energy.

[8.6].Journals : Journals like "Renewable Energy", "Solar Energy", and "Journal of Solar Energy Engineering" often publish articles on predictive modeling in solar energy.