International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 08 Issue: 04 | April - 2024 SJIF Rating: 8.448 ISSN: 2582-3930

Solar Powered Car Optimization Through Machine Learning

Bolleddu Jashwanth[1], A. Bhavya Madhuri[2], M. Akash[3], K. Madhu Babu[4]

1 *UG student, Dept. of Electronics and Computers Engineering, Sreenidhi Institute of Science and Technologies, Telangana, India*

2 *UG student, Dept. of Electronics and Computers Engineering, Sreenidhi Institute of Science and Technologies, Telangana, India*

3 *UG student, Dept. of Electronics and Computers Engineering, Sreenidhi Institute of Science and Technologies, Telangana, India*

4 *Asst. Professor, Dept. of Electronics and Computers Engineering, Sreenidhi Institute of Science and Technologies, Telangana, India*

Abstract: The main goal of the project is to predict the solar power output of the solar panel depending on prevailing weather conditions using some Machine Learning algorithms for the purpose of optimizing and adapting the power consumption so as to strike a good balance between solar power supply and demand. For predicting the power output, the following parameters are measured with the help of an Arduino-based sensor unit -Temperature, Pressure, Light intensity and Relative humidity. Apart from them as well, cloud coverage, visibility, and dew point, and we add wind speed is also taken into consideration which we collect from the weather forecast that help a lot. In order in an attempt to train the Machine Learning algorithms, a dataset using the above parameters will be formed hopefully. As renewable energy is gaining a lot of ground rapidly, it becomes necessary for us interestingly to make solar systems reliable kind of. By predicting the power output of the panel we can plan a Necessary alternative accordingly as well, making the system more reliable indubitably. Solar energy prediction is significant in enhancing the competitiveness of solar power plants in the energy market indeed, and decreasing reliance on fossil fuels in socio-economic development largely. Our work is aiming to accurately predict the solar energy.

IndexTerms–MachineLearning, SolarPower, Arduino Uno, PythonJupyterNotebook, Sensors.

1. INTRODUCTION

Solar, which is a form of renewable energy, is rapidly gaining ground like never before.This makes it even more importantly significant for us, to equip ourselves with the latest technologies that can improve the reliability of solar systems. Solar energy prediction is a key element, quite essential, in enhancing the competitiveness of solar power plants in the energy market, and decreasing reliance on fossil fuels in socio-economic development. The main aim of this project is to accurately predict the solar energy. By predicting solar power, various activities that require power consumption can be scheduled well in advance depending on their needs. If in case the weather conditions are unfavorable, resulting in inadequate power generation, alternate energy backup can be very well planned so as to avoid any disruptions in the activities. With the growing deployment of solar energy into modern grids, PV solar energy prediction has become increasingly important to deal with the volatility and uncertainty that are associated with solar power in these systems. The grid can very well use the models to plan generator dispatch schedules well in advance the use of real-time prediction allows to adjust well to changes in production and to react to complex events of exceptionally high or low load consumption. In addition, it decreases significantly the amount of operating reserves that are needed by the system thus reducing the system balancing costs.

PROBLEM STATEMENT

If any factory initiates the work using solar power, the power generated by the panels may or may not match the required power to complete a specific task in an industry. In those cases, the work may get interrupted in between before the completion of the whole tasks, it may winds to heavy loss. So, this project work aims to accurately predict the solar energy forecasting. This can be done using Machine Learning algorithms for optimizing and adapting the power consumption so as to strike a balance between solar power supply and demand.

OBJECTIVE OF THE PROJECT

To prognosticate the solar power of the solar panel depending on prevailing atmospheric conditions using Machine Learning algorithms. To prognosticate solar power, different factors are estimated using an Arduino-grounded LDR detector to measure Light Intensity, a BMP180 Detector for Pressure, a DHT11 Detector for Temperature and moisture, and a Solar panel to determine Solar power. To train the Logistic Retrogression, Decision Tree, and KNN model, a dataset using the below parameters will be prepared. As renewable energy is gaining ground fleetly, We need to make solar systems dependable. By prognosticating solar power. We can plan a necessary volition consequently, making the system more dependable.

2. EXISTING METHODS AND IT'S LITERATURE SURVEY

Past thinks about have investigated comparable destinations utilizing elective strategies. Be that as it may, to improve forecast exactness and framework flexibility, we are receiving Machine Learning calculations in this paper.

i. Past-Predicts-Future Show (PPF): Past-Predicts-Future show (PPF), which employments the past day sun based escalated to anticipate the another day sun oriented escalated. Past predicts future models are regularly utilized when estimates are not accessible, since the past could be a sensibly great pointer of the longer term on the off chance that the climate does not alter. Whereas they are exceedingly exact in the event that climate conditions don't alter, the models are not able to anticipate exceptional changes in the climate. **[Navin Sharma, Pranshu Sharma, David Irwin, Prashant Shenoy, "Predicting Solar Generation from Weather Forecasts Using Machine Learning".] [1]**

ii. The Cloudy Model: The cloudy model could be a basic demonstration that employments as it were the sky condition as premise for the forecast. The cloudy model is more exact than existing variations of PPF. It doesn't join data from numerous climate measurements and their affect on sun powered escalated. Whereas this show is able to foresee any changes in climate, it doesn't join data from numerous climate measurements and their effect on sun oriented escalated**. [Navin Sharma, Pranshu Sharma, David Irwin Prashant Shenoy, "Predicting Solar Generation from Weather Forecasts Using Machine Learning".] [1]**

iii.Least Square Linear Regression: Least Square Linear Regression could be a machine learning calculation based on supervised learning. It performs a relapse errand. This can be used to predict the exact esteem of the subordinate variable (sun based control) which may be a work of a few autonomous factors. It is less demanding to actualize, decipher and effective to prepare. Relapse models are target expectation esteem based on autonomous factors**. [Suruchi Dedgaonkar, Vishal Patil, Niraj Rathod, Gajanan Hakare & Jyotiba Bhosale, "Solar Energy Prediction using Least Square Linear Regression Method", International Journal of Current Engineering and Technology, Vol.6, No.5 (Oct 2016), E-ISSN 2277–4106, P-ISSN 234 –5161.] [3]**

3. Solar Power Output Prediction Using Machine Learning Algorithms

The methodologies such as Logistic Regression, KNN and Decision Tree are implemented.

i.Logistic regression: Logistic regressions may be a factual strategized utilized for demonstrating the relationship among a subordinate variable and one or more autonomous factors. The objective of calculated relapse are to discover the line of best fitting that represents the relationships among the factors, and can be utilized for forming predictions about the subordinate variable based on values of the free variables!

$$y = b0 + b1 * x$$

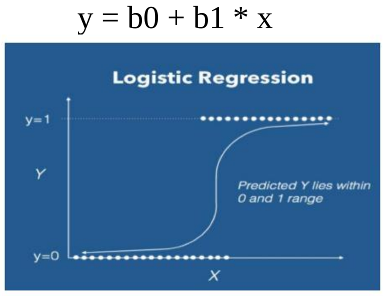

Figure 1: Logistic Regression Model

ii. Decision Tree: A decision tree may be a sort of machine learning calculation that's used for classification and relapse issues. It could be a tree-like show that comprises of an arrangement of choices or parts, spoken to by inside hubs, and results or forecasts, spoken to by leaf hubs. decision trees are basic to get it and decipher as appeared in fig 2, and they can handle both numerical and categorical factors. Here, the anticipated Sun powered Control is classified in three ways such as Moo, Medium, and Tall.

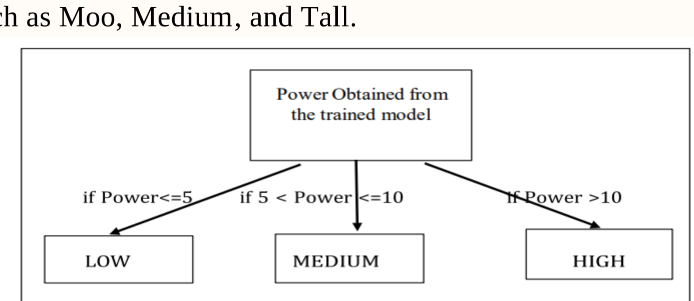

Figure 2: Decision Tree Model

iii. K-Nearest Neighbors: K-Nearest Neighbor (k-NN) is an easy but good machine learning algorithm applied for both classification and regression tasks. It is a non-parametric and case-based learning method, indicating it doesn't make strong assumptions about the basic distribution of the data and instead relies on the data instances themselves during training and prediction.

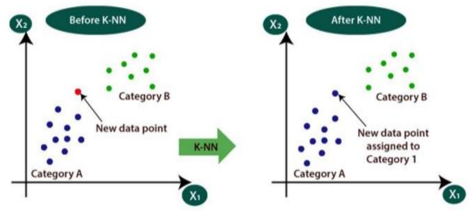

Figure 3: K Nearest Neighbor Model

4. IMPLEMENTATION OF THE SOLAR POWER OUTPUT PREDICTION USING ARDUINO BASED SENSORS AND MACHINE LEARNING MODELS

In this chapter the equipment and computer program depiction were examined together with the square graph for the preparation of dataset and clarified the reason of each square. Within the same chapter the clarification of Arduino Uno, sensor units, sun oriented board, Arduino IDE and Python Jupyter Notepad are given with usage of extend.

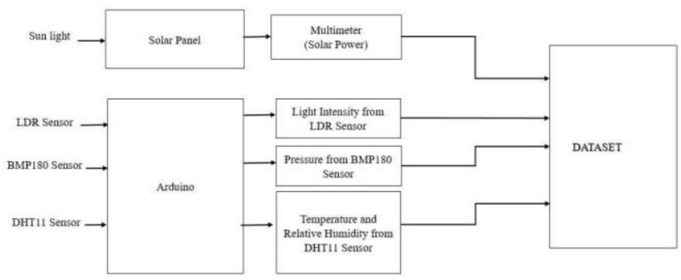

Fig 4: Block diagram for Dataset Preparation

The fig 4. shows the block diagram shows about the dataset preparation and prediction of the model based on the machine learning algorithms of the project.

I Data Set Preparation:

In this extend to begin with the dataset is ready by utilizing diverse parameters like Light Escalated from the LDR Sensor, Weight from BMP180 Sensor, Temperature and Relative Mugginess from DHT11 Sensor utilizing Arduino as appeared in piece graph fig 6. The Arduino board is modified with Arduino IDE by utilizing C dialect to degree the over parameters with their particular code as shown in figure alongside parameters the Sun based Control is measured by utilizing sun oriented board. In arrange to prepare the show for that, we employ machine Learning calculations, and a dataset utilizing over parameters will be formed. In addition, the wide run of parameters considered for planning the dataset makes our demonstration more bonafide and exact. Firstly, Sun oriented board is exposed to the sunlight and the introduction of sun powered board is additionally critical. It is to guarantee that it covers the greatest zone in uncovering to the sun. With the offer assistance of a multimeter the Current(I) and Voltage(V) are measured, at that point the control is calculated as appears within the fig 5.

Power = Voltage * Current

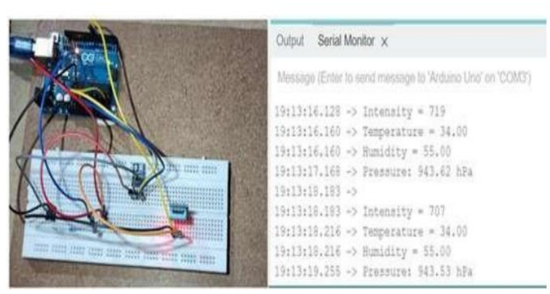

Figure 5:Measuring solar power using Multimeter From solar Panel.

Figure 6:Collecting Atmospheric data using Sensor Unit.

The sensor unit is associated with the Arduino UNO with the assistance of breadboard and interfacing wires,and interfaces with the Arduino IDE Computer program. The control from the sun-based board and information from the sensor unit are recorded simultaneously and organized to create the dataset. Presently, it is got to indicate the input and yield columns of the dataset where comparing values of the sensor are arranged in their particular columns without any bungles. The Dataset is ready in such a way that, minimization of mistakes is thought to be maintained and no copies are permitted. The dataset test which is ready for preparing the Machine Learning Models as appears within tables utilized for Calculated Replace and KNN models.

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 08 Issue: 04 | April - 2024 SJIF Rating: 8.448 ISSN: 2582-3930

1	Intensity	Humidity	Pressure	Temperatu	Power
2	928	57	1011	29	3.26
3	975	45	1012	32	13.56
4	946	49	1015	38	7.286
5	948	47	1019	41	7.45
6	943	46	1014	44	7.43
7	903	41	1014	29	3.42
8	926	35	1014	30	3.82
9	929	35	1014	30	4.26
10	944	32	1011	32	7.24
11	942	18	1010	38	8.46
12	927	18	1009	39	8.65
13	905	18	1005	38	2.36
14	860	16	1005	37	2.84
15	799	16	1005	37	3.21
16	530	17	1010	36	2.01
17	876	36	1004	29	3.13
18	871	34	1011	31	2.94
19	938	33	1010	31	3.12
20	1023	23	1009	34	14.28
21	949	20	1008	35	6.28
22	945	18	1007	35	6.21
23	930	16	1006	38	4.36
24	855	18	1010	36	4.24
25	806	59	1011	30	3.26
26	927	51	1011	32	3.65
27	940	35	1009	33	6.54
28	941	21	1007	36	7.23
29	868	21	1007	37	3.56
30	837	23	1011	35	3.21

1	Intensity	Humidity	Pressure	Temperatu	Power
2	928	57	1011	29	L
3	975	45	1012	32	H
4	946	49	1015	38	M
5	948	47	1019	41	M
6	943	46	1014	44	M
7	903	41	1014	29	L
8	926	35	1014	30	L
9	929	35	1014	30	L
10	944	32	1011	32	M
11	942	18	1010	38	M
12	927	18	1009	39	M
13	905	18	1005	38	L
14	860	16	1005	37	L
15	799	16	1005	37	L
16	530	17	1010	36	L
17	876	36	1004	29	L
18	871	34	1011	31	M
19	938	33	1010	31	L
20	1023	23	1009	34	H
21	949	20	1008	35	M
22	945	18	1007	35	M
23	930	16	1006	38	L
24	855	18	1010	36	L
25	806	59	1011	30	L
26	927	51	1011	32	L
27	940	35	1009	33	M
28	941	21	1007	36	M
29	868	21	1007	37	L

Table 1 :Data set for KNN and Logistic Regression Table 2: Data set for Decision Tree Algorithm

As both of the tables contains the data that need to be trained to the Machine Learning Algorithms that we have proposed like KNN ,Logistic Regression, Decision Tree.

5. TRAINING MACHINE LEARNING ALGORITHMS:

In this paper, an alternate step is prognosticating the model grounded on a machine literacy algorithm. In the training-phase the dataset which is set is given to the machine literacy model to train the model with the training and testing datasets split into 50 and 50 probabilities. The machine literacy models are trained by the datasets which are prepared before as they will be shown in further steps of the paper.

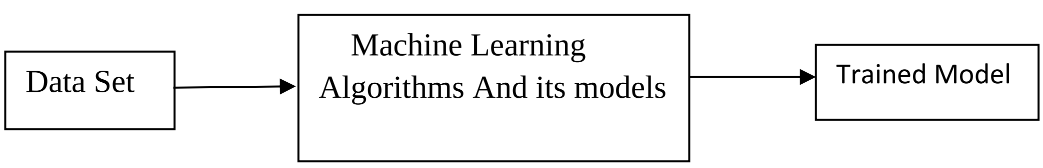

Fig 7: Training The Algorithms

Importing Required Libraries for the Machine Learning Models:

```
[113]: import numpy as np
       import pandas as pd
       import matplotlib.pyplot as plt
       from sklearn.model_selection import train_test_split
       from sklearn.linear_model import LinearRegression
       from sklearn import metrics
       from sklearn.metrics import r2_score
       from sklearn.metrics import accuracy_score
       from sklearn import preprocessing
       from sklearn.linear_model import LogisticRegression
```

Fig 8: Logistic Regression

```
[125]: import numpy as np
       import pandas as pd
       from sklearn.model_selection import train_test_split
       from sklearn import metrics
       from sklearn.metrics import r2_score
       from sklearn.neighbors import KNeighborsRegressor
       from sklearn.neighbors import KNeighborsClassifier
       from sklearn.metrics import accuracy_score
       from sklearn.metrics import confusion_matrix,classification_report,accuracy_score
```

Fig 9: K Nearest Neighbors

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 08 Issue: 04 | April - 2024 SJIF Rating: 8.448 ISSN: 2582-3930

```
[27]: import numpy as np
      import pandas as pd
      from sklearn.model_selection import train_test_split
      from sklearn import metrics
      from sklearn.metrics import r2_score
      from sklearn.tree import DecisionTreeRegressor
      from sklearn.tree import DecisionTreeClassifier
      from sklearn.metrics import accuracy_score
      from sklearn.metrics import confusion_matrix,classification_report,accuracy_score
```

Fig 10: Decision Tree

Training Machine Learning Models:

```
df = pd.read_csv(r"Majordata.csv")

[116]: x=df[['Intensity','Humidity','Pressure','Temperature']]
       y=df[['Power']]

[117]: x_train,x_test,y_train,y_test = train_test_split(x,y,test_size=0.50,random_state=50)
       linreg = LinearRegression()

[126]: model = LogisticRegression()
       model.fit(x_train, y_train.values.ravel())
```

Fig 11: Logistic Regression

```
[116]: df = pd.read_csv(r"Majordata.csv")

[117]: x=df[['Intensity','Humidity','Pressure','Temperature']]
       y=df[['Power']]

[118]: x_train,x_test,y_train,y_test = train_test_split(x,y,test_size=0.50)

[128]: knn = KNeighborsClassifier(n_neighbors=7)
       knn.fit(x_train,y_train.values.ravel())
```

Fig 12: KNN

```
[28]: df = pd.read_csv(r"Majordata.csv")

[30]: x=df[['Intensity','Humidity','Pressure','Temperature']]
      y=df[['Power']]

[31]: x_train,x_test,y_train,y_test = train_test_split(x,y,test_size=0.50)

[36]: Decision_model = DecisionTreeClassifier()

[37]: Decision_model.fit(x_train,y_train)
```

Fig 13: Decision Tree

Testing Phase of the Trained Models:

```
[128]: y_pred = model.predict(x_test)
       accuracy = accuracy_score(y_test, y_pred)
       print("Accuracy: {:.2f}%".format(accuracy * 100))
```

Fig 14: Logistic Regression

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 08 Issue: 04 | April - 2024 SJIF Rating: 8.448 ISSN: 2582-3930

```
[38]: decision_pred = Decision_model.predict(x_test)

[39]: decision_accuracy = accuracy_score(y_test,decision_pred)

[40]: decision_accuracy*100
```

```
[133]: y_pred = knn.predict(x_test)

[143]: knn.score(x_train,y_train)

[143]: 0.9254385964912281

[144]: from sklearn.metrics import confusion_matrix
       confusion_matrix(y_test,y_pred)

[144]: array([[ 13,   0,  15],
              [  0,  16,   1],
              [  2,   4, 178]], dtype=int64)

[145]: from sklearn.metrics import accuracy_score
       accuracy_score(y_test,y_pred)
```

Fig 15: Decision tree

Fig 16: KNN Algorithm

As shown in above Figures both the testing and training phases are done and now we are about to know the accuracy results that each of the Algorithms carried out by training the algorithms**.**

6. RESULTS

By Analysing the results of the trained models of every Machine learning algorithm we can clearly conclude that KNNAlgorithm have the highest accuracy rate as compared to the Logistic regression and Decision tree model. Next we can consider that Decision Tree have high accuracy but it doesn't provide the exact output of the power its is generated.

```
[38]: decision_pred = Decision_model.predict(x_test)

[39]: decision_accuracy = accuracy_score(y_test,decision_pred)

[40]: decision_accuracy*100

[40]: 89.51965065502183
```

```
[144]: from sklearn.metrics import confusion_matrix
       confusion_matrix(y_test,y_pred)

[144]: array([[ 13,   0,  15],
              [  0,  16,   1],
              [  2,   4, 178]], dtype=int64)

[145]: from sklearn.metrics import accuracy_score
       accuracy_score(y_test,y_pred)

[145]: 0.9039301310043668
```

Fig 17: Decision Tree

```
[128]: y_pred = model.predict(x_test)
       accuracy = accuracy_score(y_test, y_pred)
       print("Accuracy: {:.2f}%".format(accuracy * 100))

       Accuracy: 83.41%
```

Fig 18 : Logistic Regression

Fig 19 : KNN

So finally we can conclude that KNN have the more accuracy compared to Decision Tree and Logistic Regression model. KNN have 90% of accuracy, Decision Tree have 89% of accuracy and Logistic Regression have 83% of accuracy**.**

7. APPLICATION: SOLAR CHARGEABLE BATTERY CAR

Solar Chargeable battery car is an easy way to save income on Guests that can be twenty spent on fuel! exhilarate in the comfort of your vehicle with zero emissions and no detrimental gas! Easy charges right in your sunlight during the day and be ready to drive all have long. Save money while contributing to a cleaner crucial**.**As if we use a solar car that which charges its battery through the sunlight which is more cost affective and it's a open source of energy.

Solar Power Output will be mainly based on the Solar Panel that we are using and it is mainly based on the size of the solar panel and the output that solar panel can carry.

Fig 20: Solar Car

Our application's main aim is to know how much time will it take to charge a battery by using a solar panel with the help of sunlight. After getting the output power that we have predicted from the Machine Learning Algorithms we can know how much time we need to charge the battery to reach the required no of miles that we want to travel.
By predicting the Power output prior we can know whether how much output power we can expect for the next day by that we can plan an alternate source of energy to charge the car battery.

8.CALCULATION:

To calculate the time taken by a solar panel to charge a 3.7 Volt battery with a capacity of 1500 mAh(Milliamphere-hour **)**

$$T = C/I_{charge}$$

- **T** is the time taken to charge the battery(in hours).
- **C** is the capacity of the battery(in ampere hours,Ah).
- **I**charge is the charging current

Conversion of Battery Capacity: At first we need to convert the battery capacity from mAh to Ah.
1Ah = 1000mAh then the capacity (C) in ampere-hours is:

$$\boxed{C = {}^{1500mAh}/_{1000} = 1.5\text{Ah}}$$

$$P_{\text{solar}} = V_{\text{battery}} \times I_{\text{charge}}$$

In this case, $V_{\text{battery}} = 3.7$ volts. Therefore:

$$I_{\text{charge}} = \frac{P_{\text{solar}}}{V_{\text{battery}}} = \frac{10 \text{ watts}}{3.7 \text{ volts}} \approx 2.70 \text{ amperes}$$

Applying the formula:

Substitute the values in

$$T = \frac{C}{I_{\text{charge}}}:$$

$$T = \frac{1.5\ \text{Ah}}{2.70\ \text{A}} \approx 0.556\ \text{hours}$$

So that approximately it takes 0.556 hours it is 33.36 minutes fro a 10watt solar panel with 80% Efficiency to charge a 3.7volts 1500mAh battery.
This is how one can calculate and know how much time does it takes to charge a 3.7volts battery with 1500mAh capacity.In real life it is useful to know how much output we can expect from next day as we predict the power output with the help of Machine learning Algorithms.

9.CONCLUSION:

In this project we have used three methods to predict the actual output of the solar power from Atmospheric Parameters.By using Decision Tree, Logistic regression and K Nearest neighbor The parameter considered for goodness of the model is accuracy, complexity.
After performing this project, we have observed that KNN shows more accuracy than compared toDecision -Tree and Logistic regression.Then Decision tree show great accuracy after the KNN which is very close to Knn but the biggest draw back for decision tree is it can’t show the exact number for the power output it Only shows the power output in the form of L,M and H.
Therefore it is concluded that K Nearest Neighbor Algorithm shows the exact values of the solar power output with high accuracy compared with other two Algorithms and also it is concluded that solar power output is mainly depends on the light intensity of sun. Whenever the intensity is low then we can’t expect good power output.

10.FUTURE SCOPE

Predicting solar power is crucial for various activities that need power required by them. As solar energy is being used more and more in modern power distribution systems PV solar energy prediction is getting more important to handle the unpredictable nature of solar power in these grids. The models for it can enable the grid to schedule generator dispatching in advance, increasing the competition of solar power plants. On top of that, real-time prediction allows adjusting to production changes and coping with intricate events like excessively high or low load production or consumption. Furthermore, it decreases the requirement of operating reserves, therefore bringing down the system's balancing expenses. With real-time solar energy prediction, planners, decision-makers, and power plant operators can make quick decisions and manage the systems reliably and effectively.

11.REFERENCES:

[1] [Navin Sharma, Pranshu Sharma, David Irwin, Prashant Shenoy, “Predicting Solar Generation from Weather Forecasts Using Machine Learning”.]
[2] [Suruchi Dedgaonkar, Vishal Patil, Niraj Rathod, Gajanan Hakare & Jyotiba Bhosale, “Solar Energy Prediction using Least Square Linear Regression Method”, International Journal of Current Engineering and Technology, Vol.6, No.5 (Oct 2016), E-ISSN 2277–4106, P-ISSN 234 –5161. **]**
[3] [P. Li, K. Zhou, S. Yang, “Photovoltaic Power Forecasting: Models And Methods,” 2nd IEEE Conference on Energy Internet and Energy System Integration, EI2 2018 - Proceedings, 1–6, 2018.]
[4] [Imane Jebli, Fatima-Zahra Belouadha , Mohammed Issam Kabbaj, Amine Tilioua, “Deep Learning Based Models For Solar Energy Prediction”, Advances In Science, Technology And Engineering Systems Journal Vol. 6, No. 1, 349-355 (2021), ISSN: 2415-6698]
[5] [**Xiangyun Qing, Yugang Niu “**Hourly day-ahead solar irradiance prediction using weather forecast By LSTM” February 2018 **Energy 148** **http://dx.doi.org/10.1016/j.energy.2018.01.177**]
[6] **[**P. Li, K. Zhou, S. Yang, “Photovoltaic Power Forecasting: Models And Methods,” 2nd IEEE Conference on Energy Internet and Energy System Integration, EI2 2018 - Proceedings, 1–6, 2018.]
[7] [Babu Ram ,Dileep Singh Rathore , Sunil Malviya and B.L. Pal, “Analysis of Classification Algorithms for Insect Detection using MATLAB”, 2nd INTERNATIONAL CONFERENCE ON ADVANCED COMPUTING AND SOFTWARE ENGINEERING (ICACSE-2019).]
[8] [https://www.researchgate.net/publication/358905399_Prediction_of_Solar_Power_Using_Machine Learning_Algorithm]
[9] [https://www.researchgate.net/publication/366613171_Solar_Power_Prediction_using_Regression_ Models**, Mustafa Yasin Erten, Hüseyin AYDİLEK]**

[10] [Solar Energy Prediction using Least Square Linear Regression Method **Suruchi Dedgaonkar**,Vishal patil, Niraj Rathod, Gajanan Hakare October 2016]