SOLAR POWER FORECASTING USING MACHINE LEARNING

B.Tech, 6th Semester Examination, 2024

Paper Name: Minor Project II

Paper Code: PR 691

Submitted By:

Shayak Chakraborty (University Roll No. - 501021010014)

Tomoit Ghosh (University Roll No.- 501021010017)

Arpan Das (University Roll No. - 501021010001**)**

Guided By:

Ms. Suparna Maity
(Assistant Professor)
Department of Electronics and Computer Science

Department of Electronics and Computer Science

GURU NANAK INSTITUTE OF TECHNOLOGY

Affiliated By

Maulana Abul Kalam Azad University of Technology

GURUNANAK INSTITUTE OF TECHNOLOGY

157/F, Nilgunj Rd, Sahid Colony, Panihati, Kolkata, West Bengal 700114

Affiliated by:

MAULANA ABUL KALAM AZAD UNIVERSITY OF TECHNOLOGY

This is to certify that the project work titled **"SOLAR POWER FORECASTING USING MACHINE LEARNING"** has been carried out by *Shayak Chakraborty, Tomojit Ghosh* and *Arpan Das* under my supervision in fulfillment of the requirements for the degree of Bachelor of Technology in Electronics & Computer Science Engineering of the MAKAUT during the academic year 2023-2024.

_____

Ms. Suparna Maity

(Assistant Professor, Dept. of ECS)

_____

COUNTERSIGNED

Ms. Bapita Roy

Head of the Department- Department of ECS

GURU NANAK INSTITUTE OF TECHNOLOGY

157/F, Nilgunj Rd, Sahid Colony, Panihati, Kolkata, West Bengal 700114

Affiliated by:

MAULANA ABUL KALAM AZAD UNIVERSITY OF TECHNOLOGY

CERTIFICATE OF APPROVAL

The following project report is hereby approved as a creditable study of an Engineering subject carried out and presented in a manner satisfactory to warrantits acceptance as a pre-requisite to the degree for which it has been submitted. Itis understood that by this approval the undersigned do not necessarily endorse or approve any statement made, opinion expressed or conclusion drawn therein but approve the project only for the purpose for which it is submitted.

EXAMINERS

_____

_____

_____

ACKNOWLEDGEMENT

We express our deepest sense of gratitude to our respected and learned guide*, Ms. Suparna Maity* for her valuable help and guidance. I'm also thankful to all my group members *Tomojit Ghosh* & *Arpan Das* for the encouragement they have given in completing the project.

We are thankful to the Head of the Department, *Ms. Bapita Roy* for permitting us to utilize the necessary facilities of the institution.

We are also thankful to all other faculty staff members of our department for their kind cooperation and help.

_____

_____

_____

Contents

Chapter 1

Introduction

1.1 Introduction

Solar power forecasting is a critical component in the integration of renewable energy into the electrical grid. As solar energy becomes a more significant part of the energy mix, accurately predicting solar power generation is essential for ensuring grid stability, optimizing energy storage, and planning energy dispatch. The inherent variability of solar energy, due to factors such as weather conditions, atmospheric phenomena, and seasonal changes, presents significant challenges for reliable forecasting. Recent advancements in machine learning (ML) have shown great promise in addressing these challenges. Machine learning algorithms can analyze vast amounts of historical and real-time data to identify patterns and make precise predictions. Unlike traditional statistical methods, machine learning models can capture complex, nonlinear relationships between input variables and solar power output, enhancing forecasting accuracy.

It highlights real-world applications and case studies where machine learning has been successfully implemented for solar power forecasting, demonstrating the tangible benefits of this approach. By harnessing the power of machine learning, solar power forecasting can achieve greater precision, leading to more effective grid management, reduced reliance on fossil fuels, and the promotion of sustainable energy practices.

It explores the application of machine learning techniques in solar power forecasting. It covers various machine learning methodologies, including supervised learning, unsupervised learning, and ensemble methods, and discusses their effectiveness in predicting solar irradiance and power generation.

By leveraging machine learning, solar power forecasting can achieve higher precision, thereby contributing to more efficient grid management and facilitating the broader adoption of solar energy. This, in turn, supports the transition to a more sustainable and resilient energy system.

1.2 Objective

It aims to highlight the importance of accurate solar power forecasting in integrating renewable energy into the electrical grid and overcoming the challenges of solar energy variability. The report will explore various machine learning methodologies, such as supervised learning, unsupervised learning, and ensemble methods, and explain their benefits for predicting solar irradiance and power generation. Key components of the forecasting process, including data collection, feature selection, data preprocessing, and model evaluation, will be discussed to emphasize the development of robust models. Additionally, real-world applications and case studies will be reviewed to demonstrate the practical benefits of machine learning in this field. It will assess the advantages and challenges of using machine learning over traditional methods and provide recommendations for future research and development to enhance solar power forecasting.

1.3 Literature Survey

- Short-Term Solar Power Forecasting Based on Machine Learning Techniques: A Review by S. Zhang et al. (*Renewable and Sustainable Energy Reviews, 2019*)

This review paper provides a comprehensive overview of machine learning techniques used for short-term solar power forecasting. It covers various models, such as support vector regression, artificial neural networks, and hybrid models, discussing their strengths and weaknesses.

- Solar Power Prediction Using Data Analytics: A Review by R. Gupta et al. (*Renewable and Sustainable Energy Reviews, 2017*)

This review paper offers an overview of data analytics techniques used for solar power prediction, including statistical models, machine learning models, and artificial neural networks. It also covers the different data sources used for solar power prediction, such as meteorological data and satellite imagery.

- Solar Power Forecasting Using Artificial Neural Networks: A Review by S. Bhowmik et al. (*Renewable and Sustainable Energy Reviews, 2020*)

This review paper focuses on the use of artificial neural networks for solar power forecasting. It covers various types of neural networks, such as feedforward neural networks, recurrent neural networks, and convolutional neural networks, and discusses their applications in solar power prediction.

- Review of Solar Power Forecasting Methodologies by N. Shrestha et al. (*Renewable and Sustainable Energy Reviews, 2019*)

This review paper provides an overview of solar power forecasting methodologies, including statistical models, machine learning models, and hybrid models. It also covers the different data sources used for solar power prediction and discusses the challenges and opportunities in solar power forecasting.

- Machine Learning for Solar Energy Prediction: A Review by A. S. Mohan et al. (*Renewable and Sustainable Energy Reviews, 2021*)

This review paper provides an overview of machine learning techniques used for solar energy prediction, including regression models, artificial neural networks, and decision trees. It also discusses the challenges and opportunities in solar energy prediction and provides a perspective on future research directions.

Chapter 2

Flow Chart

2.1 Flow Chart

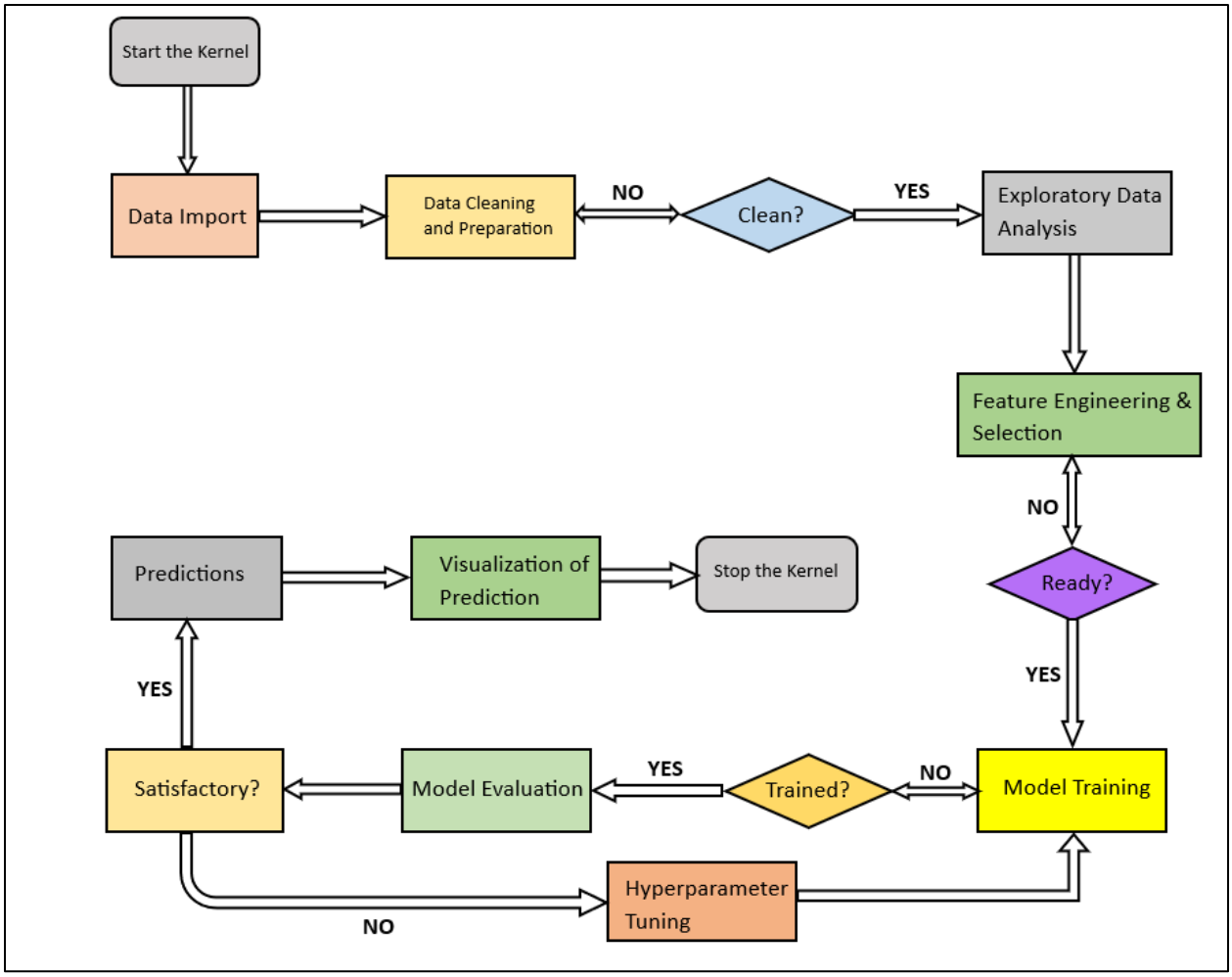

Fig 1: Flow Chart of the entire model

2.2 Description of the Flow Chart

Data Import: The project begins with importing the dataset containing relevant features for solar power forecasting.

Data Cleaning and Preparation: The data undergoes cleaning and preparation, which includes handling missing values, dealing with outliers, and ensuring data quality.

Clean? (Decision Point): A check is performed to ensure the data is clean. If the data is not clean, it returns to the cleaning and preparation step.

Exploratory Data Analysis (EDA): Once the data is clean, EDA is conducted to understand the underlying patterns, distributions, and relationships within the data.

Feature Engineering & Selection: Relevant features are engineered and selected based on insights from the EDA to enhance the model's performance.

Ready? (Decision Point): A check is performed to ensure the features are ready for model training. If not, it goes back to the feature engineering step.

Model Training: The model is trained using the prepared features. Various regression algorithms are tested to find the best performing model.

Trained? (Decision Point): A check is performed to determine if the model is adequately trained. If not, hyperparameter tuning is performed.

Hyperparameter Tuning: Hyperparameters of the model are adjusted to optimize its performance and improve accuracy.

Model Evaluation: The trained model is evaluated to assess its performance using metrics like RMSE. Visualization of predictions helps in understanding the model's accuracy.

Satisfactory? (Decision Point): A decision is made whether the model's performance is satisfactory. If not, the process returns to hyperparameter tuning.

Predictions: Once the model is satisfactory, it is used to make predictions on new, unseen data.

Visualization of Prediction: The predictions are visualized to provide insights and validate the model's performance.

Chapter 3

Correlation Heatmap

3.1 Heatmap

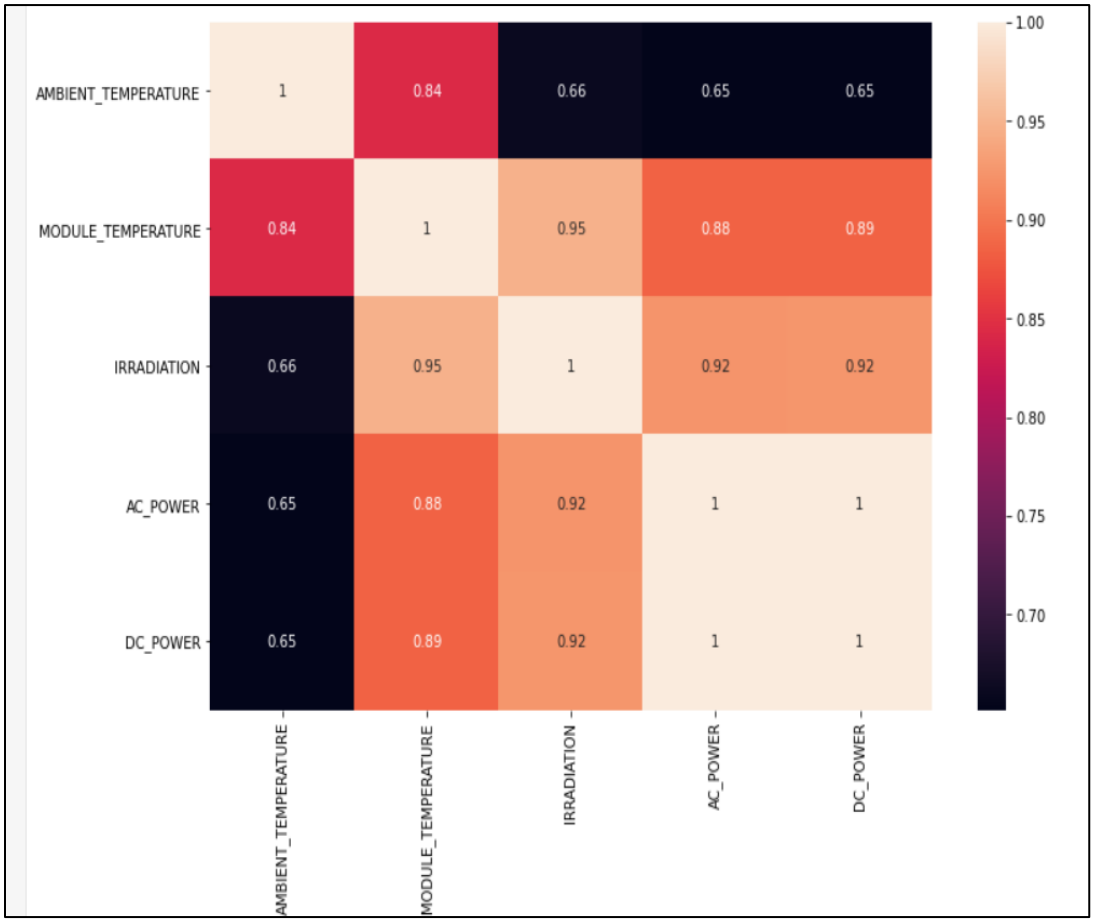

Fig 2: Heatmap of the model

3.2 Heatmap's Variables

The heatmap illustrates the correlation coefficients between key variables in the dataset. Correlation values range from -1 to 1, where 1 indicates a perfect positive correlation, -1 indicates a perfect negative correlation, and 0 indicates no correlation.

Variables are Ambient Temperature, Module Temperature, Irradiation, AC and DC Power.

3.3 Description of the Variables

1. **Ambient Temperature and Module Temperature:** Strong positive correlation (0.84), indicating that as ambient temperature increases, module temperature also tends to increase.

2. **Irradiation and Module Temperature:** Very strong positive correlation (0.95), suggesting higher solar irradiation significantly increases module temperature.

3. **Irradiation and AC/DC Power:** Both have very strong positive correlations (0.92) with irradiation, implying that higher solar energy results in higher power generation.

4. **AC Power and DC Power:** Perfect correlation (1.0), showing that AC power and DC power are directly proportional and vary together perfectly.

Chapter 4
Model and Simulation

4.1 Algorithm

Step 1: Data Import: Import the Plant Generation and Weather sensor dataset containing solar power-related features such as ambient temperature, module temperature, irradiation, AC and DC power, Date and Time.

Step 2: Data Cleaning: Handle missing values using appropriate imputation methods. Detect and handle outliers to ensure data quality. Ensure consistency and integrity of the dataset.

Step 3: Exploratory Data Analysis: Perform descriptive statistics to summarize the data. Create visualizations (e.g., histograms, scatter plots, heatmaps) to understand data distributions and relationships. Conduct correlation analysis to identify significant features.

Step 4: Data Preprocessing:

- **Feature Engineering:** Create new features based on domain knowledge and insights from EDA.
- **Feature Selection:** Select relevant features using methods such as correlation analysis and feature importance scores.
- **Data Transformation:** Scale and normalize features to ensure uniformity. Encode categorical variables if present.

Step 5: Train-Test Split: Split the merged dataset into training and testing sets, typically using an 80-20 or 70-30 ratio.

Step 6: Missing Value Imputation: Apply techniques to fill in any missing values in the training and test sets.

Step 7: Model Building: Train multiple regression models, such as: Linear Regression, Decision Tree Regressor, Random Forest Regressor, Ridge Regressor, Lasso Regressor, XG Boost Regressor, Artificial Neural Network (ANN) Regressor. Use cross-validation to evaluate model performance and prevent overfitting. Random Forest has been chosen.

Step 8: Hyperparameter Tuning: Optimize model hyperparameters using Grid Search or Random Search combined with cross-validation.

Step 9: Model Evaluation: Evaluate models using metrics like Mean Absolute Error (MAE), Mean Squared Error (MSE), and Root Mean Squared Error (RMSE). RMSE has been chosen.

Step 10: Predictions: Use the selected model to make predictions on new, unseen data. Visualize the predictions and compare them with actual values to assess performance.

Step 11: Model Deployment: Deploy the final model to a production environment for real-time forecasting. Monitor the model's performance and update it as necessary with new data.

4.2 Data Inputs

Plant Generation Dataset:

Table 1 : Plant Generation dataset's head

	DATE_TIME	PLANT_ID	SOURCE_KEY	DC_POWER	AC_POWER	DAILY_YIELD	TOTAL_YIELD
0	2020-05-15 00:00:00	4136001	4UPUqMRk7TRMgml	0.0	0.0	9425.000000	2.429011e+06
1	2020-05-15 00:00:00	4136001	81aHJ1q11NBPMrL	0.0	0.0	0.000000	1.215279e+09
2	2020-05-15 00:00:00	4136001	9kRcWv60rDACzjR	0.0	0.0	3075.333333	2.247720e+09
3	2020-05-15 00:00:00	4136001	Et9kgGMDl729KT4	0.0	0.0	269.933333	1.704250e+06
4	2020-05-15 00:00:00	4136001	IQ2d7wF4YD8zU1Q	0.0	0.0	3177.000000	1.994153e+07

Weather Sensor dataset:

Table 2: Weather Sensors dataset's head

	DATE_TIME	PLANT_ID	SOURCE_KEY	AMBIENT_TEMPERATURE	MODULE_TEMPERATURE	IRRADIATION
0	2020-05-15 00:00:00	4136001	iq8k7ZNt4Mwm3w0	27.004764	25.060789	0.0
1	2020-05-15 00:15:00	4136001	iq8k7ZNt4Mwm3w0	26.880811	24.421869	0.0
2	2020-05-15 00:30:00	4136001	iq8k7ZNt4Mwm3w0	26.682055	24.427290	0.0
3	2020-05-15 00:45:00	4136001	iq8k7ZNt4Mwm3w0	26.500589	24.420678	0.0
4	2020-05-15 01:00:00	4136001	iq8k7ZNt4Mwm3w0	26.596148	25.088210	0.0

4.3 Time based Train Test Split:

Table 3: Df_train data

	BLOCK	DATE	TIME	AMBIENT_TEMPERATURE	MODULE_TEMPERATURE	IRRADIATION	DC_POWER_1	AC_POWER_1	Inverter_No_1	DC_POWER_2
0	1	2020-05-15	00:00	27.004764	25.060789	0.0	0.0	0.0	1.0	0.0
1	2	2020-05-15	00:15	26.880811	24.421869	0.0	0.0	0.0	1.0	0.0
2	3	2020-05-15	00:30	26.682055	24.427290	0.0	0.0	0.0	1.0	0.0
3	4	2020-05-15	00:45	26.500589	24.420678	0.0	0.0	0.0	1.0	0.0
4	5	2020-05-15	01:00	26.596148	25.088210	0.0	0.0	0.0	1.0	0.0
...	...	...	...	...	...	...	...	...	...	...
2966	92	2020-06-14	22:45	24.185657	22.922953	0.0	0.0	0.0	1.0	0.0
2967	93	2020-06-14	23:00	24.412542	23.356136	0.0	0.0	0.0	1.0	0.0
2968	94	2020-06-14	23:15	24.652915	23.913763	0.0	0.0	0.0	1.0	0.0
2969	95	2020-06-14	23:30	24.702391	24.185130	0.0	0.0	0.0	1.0	0.0
2970	96	2020-06-14	23:45	24.534757	23.921971	0.0	0.0	0.0	1.0	0.0

2971 rows × 72 columns

Table 4: Df_test data

	BLOCK	DATE	TIME	AMBIENT_TEMPERATURE	MODULE_TEMPERATURE	IRRADIATION	AC_POWER
2971	1	2020-06-15	00:00	24.486876	23.846251	0.0	0.0
2972	2	2020-06-15	00:15	24.509378	23.902851	0.0	0.0
2973	3	2020-06-15	00:30	24.605338	24.172737	0.0	0.0
2974	4	2020-06-15	00:45	24.679791	24.459142	0.0	0.0
2975	5	2020-06-15	01:00	24.636373	24.380419	0.0	0.0
...	...	...	...	...	...	...	...
3254	92	2020-06-17	22:45	23.511703	22.856201	0.0	0.0
3255	93	2020-06-17	23:00	23.482282	22.744190	0.0	0.0
3256	94	2020-06-17	23:15	23.354743	22.492245	0.0	0.0
3257	95	2020-06-17	23:30	23.291048	22.373909	0.0	0.0
3258	96	2020-06-17	23:45	23.202871	22.535908	0.0	0.0

288 rows × 7 columns

4.4 Model Building:

4.4.1 Model Algorithm:

Steps 1: Split Data into k-Folds:

- Shuffle and split df_train into k equal-sized folds.
- Initialize an empty list "rmse_scores" to store RMSE scores for each model.

Step 2: Initialize an empty list "model_rmse_scores" to store RMSE scores for each iteration of k-fold cross-validation.

- For each fold from 1 to k - Set the current fold as the validation set. Combine the remaining k-1 folds to form the training set.
- Split Data: xtrain and ytrain are used for variable for training. xvalid and yvalid are for validation.
- Standardize Data: After fitting a Standard Scaler on xtrain and transforming both xtrain and xvalid.
- Train the Model: Fit the model on the standardized xtrain and ytrain.
- Predict and Evaluate: Predict yvalid using the trained model. Calculate the RMSE between predicted and actual yvalid.

Step 3: Calculate Mean RMSE: Compute the mean RMSE for the model from model_rmse_scores. Append the mean RMSE to rmse_scores.

Step 4: Compare Result: Compare the mean RMSE scores for each model to identify the best performing model.

4.4.2 Algorithm Selection:

Linear Regression: A simple yet effective method for predicting continuous variables, suitable for modeling the relationship between solar irradiation and power generation.

Decision Trees: Can capture non-linear relationships and interactions between variables.

Random Forests: An ensemble method that improves prediction accuracy by averaging multiple decision trees.

Support Vector Machines (SVM): Effective for regression tasks with high-dimensional data.

Neural Networks: Can model complex relationships and interactions between features, useful for capturing non-linear dependencies in the data.

4.4.3 Choosing RMSE over MAE and MSE:

Table 5: Differences between MAE, MSE and RMSE

Metric	Formula	Description	Use Case
MAE	$\frac{1}{n}\sum_{i=1}^{n}\lvert y_i^{real} - y_i^{pred}\rvert$	Measures the average magnitude of errors in a set of predictions, without considering their direction	Used to evaluate the accuracy of regression models
MSE	$\frac{1}{n}\sum_{i=1}^{n}(y_i^{real} - y_i^{pred})^2$	Average of squared errors between actual and predicted values	Penalizes larger errors more than MAE
RMSE	$\sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i^{real} - y_i^{pred})^2}$	Square root of the average of squared errors	Further amplifies the impact of larger errors, preferred for this problem

4.5 Model Training

Using Scikit-learn Pipelines: Utilizing Scikit-learn's pipeline functionality to streamline the training process. This allows for easy management and comparison of multiple models.

Regression Algorithms: Training seven different regression algorithms with default parameters, including a 3-layered Neural Network regressor.

Standardization: Before training, we standardize the features to ensure that all variables are on the same scale.

4.6 Hyperparameter Tuning using Randomized Search

Hyperparameters are settings that need to be tuned to achieve the best possible performance from a model. For Random Forest Regressor, we focused on optimizing the following hyperparameters:

Number of Trees (n_estimators): Represents the number of trees in the forest.

Maximum Depth of Trees (max_depth): Controls the maximum depth of each tree.

Minimum Samples Split (min_samples_split): Defines the minimum number of samples required to split an internal node. Higher values prevent the model from learning overly specific patterns.

Minimum Samples Leaf (min_samples_leaf): The minimum number of samples required to be at a leaf node.

Maximum Features (max_features): Specifies the number of features to consider when looking for the best split.

Criterion: MSE (Mean Squared Error): Measures the average of the squares of the errors or deviations, which is useful for regression tasks.

```
RandomizedSearchCV(estimator=RandomForestRegressor(), n_iter=100, n_jobs=-1,
                   param_distributions={'criterion': ['mse'],
                                        'max_depth': [10, 120, 230, 340, 450,
                                                      560, 670, 780, 890,
                                                      1000],
                                        'max_features': ['auto', 'sqrt',
                                                         'log2'],
                                        'min_samples_leaf': [1, 2, 4, 6, 8],
                                        'min_samples_split': [2, 5, 10, 14],
                                        'n_estimators': [100, 500, 900, 1100,
                                                         1500]},
                   random_state=100, verbose=2)
```

Fig 3: Optimization using Randomized Search

Chapter 5

Results

5.1 Model Performance Comparison

Table 6: RMSE among different models

Regressor	RMSE (kW)
Linear Regression	2.3738
Decision Tree Regressor	2.3599
Random Forest Regressor	1.7387
Ridge Regressor	2.3774
Lasso Regressor	2.7925
XG Boost Regressor	1.8680
ANN Regressor	3.3768

Random Forest Regressor has performed the best so we are going to predict the data with Random Forest Regressor.

5.2 Predictions

After taking last 3 day's data for testing and removing all the data leakage we have predicted the RMSE for test data.

Input:

```
#Splitting into x & y
x_test = df_test[['AMBIENT_TEMPERATURE', 'MODULE_TEMPERATURE',
       'IRRADIATION']]
y_test = df_test[['AC_POWER']]

#Predicting for x_test
y_pred_rf = rf_model.predict(x_test)

print(f'Root Mean Squared Error for Test Data: {np.sqrt(mean_squared_error(y_test, y_pred_rf))}')
```

Fig 4: Code to predict the RMSE difference

Output:

METRICS	VALUE (kW)
Test RMSE	1.86830155234851

Though, the RMSE on Test set is slightly higher than what we had for Training data(1.7386 kW) but seems good considering the less amount of training data**.**

5.3 Actual vs Predicted data plot

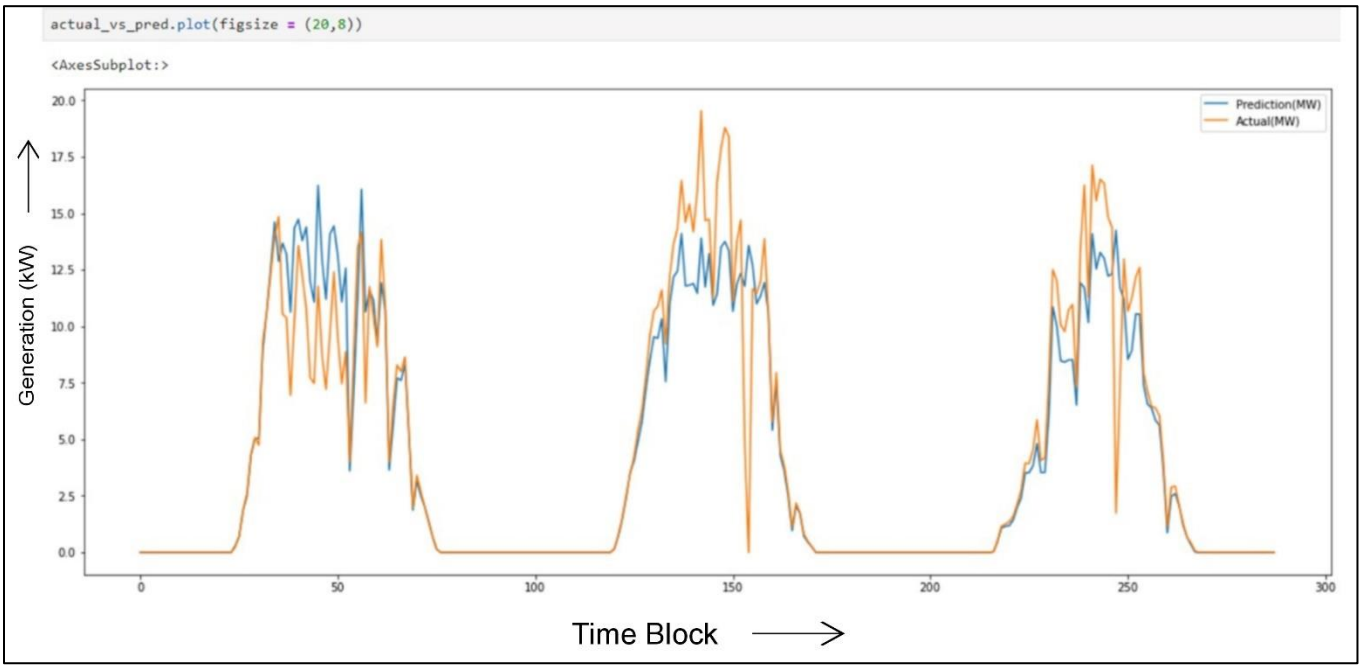

Fig 5: Actual vs Predicted Data Plot

Chapter 6

Uses & Applications

6.1 Applications

Energy Production Optimization: Accurate forecasting helps in optimizing the operation of solar power plants by predicting power generation and aligning it with demand.

Grid Stability: Reliable solar power forecasts assist grid operators in maintaining stability by effectively integrating solar power with other energy sources.

Cost Savings: Better forecasts can reduce the need for backup power and lower operational costs, leading to economic benefits for both utilities and consumers.

Maintenance Planning: Predictive insights can inform maintenance schedules, ensuring solar panels and equipment are serviced during low-production periods, thus maximizing uptime.

Policy and Planning: Governments and energy planners can use forecasting data to develop and implement policies for sustainable energy growth and infrastructure development.

Market Trading: Energy traders can leverage accurate forecasts to make informed decisions in energy markets, optimizing financial returns.

Consumer Awareness: Providing consumers with forecast data can promote energy-saving practices by aligning energy consumption with peak solar production times.

6.2 Future Scope

- Integrating of IoT devices can enhance the accuracy, efficiency, and real-time capabilities of our forecasting model.
- IoT enables remote management, anomaly detection, and predictive maintenance of solar systems.
- High-frequency data from IoT devices improves the model's ability to capture short-term fluctuations.
- IoT-assisted forecasts help balance energy supply and demand dynamically within smart grids.

6.3 Limitations

Data Quality and Availability: The accuracy of forecasts is highly dependent on the quality and completeness of historical and real-time data. Missing or inaccurate data can significantly impact model performance.

Weather Dependency: Solar power generation is heavily influenced by weather conditions, which can be unpredictable and variable. Sudden changes in weather can lead to forecasting errors.

Overfitting: Complex models might overfit to historical data, performing well on past data but poorly on new, unseen data. Proper cross-validation and regular model updates are necessary to mitigate this risk.

External Factors: Factors such as dust, shading, and equipment malfunctions are not always predictable and can affect the accuracy of solar power forecasts

Chapter 7
References

7.1 References

7.1.1 Books

[1] ”Solar Energy: The Physics and Engineering of Photovoltaic Conversion, Technologies and Systems"

Klaus Jäger, Olindo Isabella, Arno Smets, Rene van Swaaij, Miro Zeman

[2] ”Machine Learning and Data Science in the Power Generation Industry: Best Practices, Tools, and Case Studies"

Patrick Bangert

[3] ”Data Science for Supply Chain Forecasting"

Nicolas Vandeput

7.1.2 Journals

[1] ”A Comprehensive Review on Solar Power Forecasting"

U. A. Amam et al. , Renewable and Sustainable Energy Reviews, Journal of Machine Learning, volume 1, 2019

[2] “Data Preprocessing for Machine Learning in Solar Energy Forecasting"

T. Chen et al. , Renewable and Sustainable Energy Reviews, Journal of Machine Learning, volume 1, 2019

7.1.3 Datasets

https://www.kaggle.com/datasets/anikannal/solar-power-generation-data

PLAGIARISM SCAN REPORT

Date	June 12, 2024
Exclude URL:	NO

Unique Content	**87**	Word Count	704
Plagiarized Content	**13**	Records Found	0

CONTENT CHECKED FOR PLAGIARISM:

Solar power forecasting is a critical component in the integration of renewable energy into the electrical grid. As solar energy becomes a more significant part of the energy mix, accurately predicting solar power generation is essential for ensuring grid stability, optimizing energy storage, and planning energy dispatch. The inherent variability of solar energy, due to factors such as weather conditions, atmospheric phenomena, and seasonal changes, presents significant challenges for reliable forecasting. Recent advancements in machine learning (ML) have shown great promise in addressing these challenges. Machine learning algorithms can analyze vast amounts of historical and real-time data to identify patterns and make precise predictions. Unlike traditional statistical methods, machine learning models can capture complex, nonlinear relationships between input variables and solar power output, enhancing forecasting accuracy.

It highlights real-world applications and case studies where machine learning has been successfully implemented for solar power forecasting, demonstrating the tangible benefits of this approach. By harnessing the power of machine learning, solar power forecasting can achieve greater precision, leading to more effective grid management, reduced reliance on fossil fuels, and the promotion of sustainable energy practices.

It explores the application of machine learning techniques in solar power forecasting. It covers various machine learning methodologies, including supervised learning, unsupervised learning, and ensemble methods, and discusses their effectiveness in predicting solar irradiance and power generation.

By leveraging machine learning, solar power forecasting can achieve higher precision, thereby contributing to more efficient grid management and facilitating the broader adoption of solar energy. This, in turn, supports the transition to a more sustainable and resilient energy system.

1.2 Objective

It aims to highlight the importance of accurate solar power forecasting in integrating renewable energy into the

electrical grid and overcoming the challenges of solar energy variability. The report will explore various machine learning methodologies, such as supervised learning, unsupervised learning, and ensemble methods, and explain their benefits for predicting solar irradiance and power generation. Key components of the forecasting process, including data collection, feature selection, data preprocessing, and model evaluation, will be discussed to emphasize the development of robust models. Additionally, real-world applications and case studies will be reviewed to demonstrate the practical benefits of machine learning in this field. It will assess the advantages and challenges of using machine learning over traditional methods and provide recommendations for future research and development to enhance solar power forecasting.

1.3 Literature Survey

• Short-Term Solar Power Forecasting Based on Machine Learning Techniques: A Review by S. Zhang et al. (Renewable and Sustainable Energy Reviews, 2019)

This review paper provides a comprehensive overview of machine learning techniques used for short-term solar power forecasting. It covers various models, such as support vector regression, artificial neural networks, and hybrid models, discussing their strengths and weaknesses.

• Solar Power Prediction Using Data Analytics: A Review by R. Gupta et al. (Renewable and Sustainable Energy Reviews, 2017)

This review paper offers an overview of data analytics techniques used for solar power prediction, including statistical models, machine learning models, and artificial neural networks. It also covers the different data sources used for solar power prediction, such as meteorological data and satellite imagery.

• Solar Power Forecasting Using Artificial Neural Networks: A Review by S. Bhowmik et al. (Renewable and Sustainable Energy Reviews, 2020)

This review paper focuses on the use of artificial neural networks for solar power forecasting. It covers various types of neural networks, such as feedforward neural networks, recurrent neural networks, and convolutional neural networks, and discusses their applications in solar power prediction.

• Review of Solar Power Forecasting Methodologies by N. Shrestha et al. (Renewable and Sustainable Energy Reviews, 2019)

This review paper provides an overview of solar power forecasting methodologies, including statistical models, machine learning models, and hybrid models. It also covers the different data sources used for solar power prediction and discusses the challenges and opportunities in solar power forecasting.

• Machine Learning for Solar Energy Prediction: A Review by A. S. Mohan et al. (Renewable and Sustainable Energy Reviews, 2021)

This review paper provides an overview of machine learning techniques used for solar energy prediction,

including regression models, artificial neural networks, and decision trees. It also discusses the challenges and opportunities in solar energy prediction and provides a perspective on future research directions.

MATCHED SOURCES:

SOLAR POWER PREDICTION USING MACHINE ...

https://www.studocu.com/in/document/pondicherry-university/c..... (https://www.studocu.com/in/document/pondicherry-university/computer-programming/solar-power/78056271)

SOLAR POWER PREDICTION USING MACHINE ...

https://www.studocu.com/in/document/pondicherry-university/c..... (https://www.studocu.com/in/document/pondicherry-university/computer-programming/solar-power/78056271)

Check Plagiarism

PLAGIARISM SCAN REPORT

Date	June 12, 2024
Exclude URL:	NO

Unique Content	**96**	Word Count	944
Plagiarized Content	**4**	Records Found	0

CONTENT CHECKED FOR PLAGIARISM:

2.1 Flow Chart

2.2 Description of the Flow Chart

Data Import: The project begins with importing the dataset containing relevant features for solar power forecasting.

Data Cleaning and Preparation: The data undergoes cleaning and preparation, which includes handling missing values, dealing with outliers, and ensuring data quality.

Clean? (Decision Point): A check is performed to ensure the data is clean. If the data is not clean, it returns to the cleaning and preparation step.

Exploratory Data Analysis (EDA): Once the data is clean, EDA is conducted to understand the underlying patterns, distributions, and relationships within the data.

Feature Engineering & Selection: Relevant features are engineered and selected based on insights from the EDA to enhance the model's performance.

Ready? (Decision Point): A check is performed to ensure the features are ready for model training. If not, it goes back to the feature engineering step.

Model Training: The model is trained using the prepared features. Various regression algorithms are tested to find the best performing model.

Trained? (Decision Point): A check is performed to determine if the model is adequately trained. If not, hyperparameter tuning is performed.

Hyperparameter Tuning: Hyperparameters of the model are adjusted to optimize its performance and improve accuracy.

Model Evaluation: The trained model is evaluated to assess its performance using metrics like RMSE. Visualization of predictions helps in understanding the model's accuracy.

Satisfactory? (Decision Point): A decision is made whether the model's performance is satisfactory. If not, the process returns to hyperparameter tuning.

Predictions: Once the model is satisfactory, it is used to make predictions on new, unseen data.

Visualization of Prediction: The predictions are visualized to provide insights and validate the model's performance.

Chapter 3

Correlation Heatmap

3.1 Heatmap

3.2 Heatmap’s Variables

The heatmap illustrates the correlation coefficients between key variables in the dataset. Correlation values range from -1 to 1, where 1 indicates a perfect positive correlation, -1 indicates a perfect negative correlation, and 0 indicates no correlation.

Variables are Ambient Temperature, Module Temperature, Irradiation, AC and DC Power.

3.3 Description of the Variables

1. Ambient Temperature and Module Temperature: Strong positive correlation (0.84), indicating that as ambient temperature increases, module temperature also tends to increase.

2. Irradiation and Module Temperature: Very strong positive correlation (0.95), suggesting higher solar irradiation significantly increases module temperature.

3. Irradiation and AC/DC Power: Both have very strong positive correlations (0.92) with irradiation, implying that higher solar energy results in higher power generation.

4. AC Power and DC Power: Perfect correlation (1.0), showing that AC power and DC power are directly proportional and vary together perfectly.

Chapter 4

Model and Simulation

4.1 Algorithm

Step 1: Data Import: Import the Plant Generation and Weather sensor dataset containing solar power-related features such as ambient temperature, module temperature, irradiation, AC and DC power, Date and Time.

Step 2: Data Cleaning: Handle missing values using appropriate imputation methods. Detect and handle outliers to ensure data quality. Ensure consistency and integrity of the dataset.

Step 3: Exploratory Data Analysis: Perform descriptive statistics to summarize the data. Create visualizations (e.g., histograms, scatter plots, heatmaps) to understand data distributions and relationships. Conduct correlation analysis to identify significant features.

Step 4: Data Preprocessing:

• Feature Engineering: Create new features based on domain knowledge and insights from EDA.

• Feature Selection: Select relevant features using methods such as correlation analysis and feature importance scores.

• Data Transformation: Scale and normalize features to ensure uniformity. Encode categorical variables if present.

Step 5: Train-Test Split: Split the merged dataset into training and testing sets, typically using an 80-20 or 70-30 ratio.

Step 6: Missing Value Imputation: Apply techniques to fill in any missing values in the training and test sets.

Step 7: Model Building: Train multiple regression models, such as: Linear Regression, Decision Tree Regressor, Random Forest Regressor, Ridge Regressor, Lasso Regressor, XG Boost Regressor, Artificial Neural Network (ANN) Regressor. Use cross-validation to evaluate model performance and prevent overfitting. Random Forest has been chosen.

Step 8: Hyperparameter Tuning: Optimize model hyperparameters using Grid Search or Random Search combined with cross-validation.

Step 9: Model Evaluation: Evaluate models using metrics like Mean Absolute Error (MAE), Mean Squared Error (MSE), and Root Mean Squared Error (RMSE). RMSE has been chosen.

Step 10: Predictions: Use the selected model to make predictions on new, unseen data. Visualize the predictions and compare them with actual values to assess performance.

Step 11: Model Deployment: Deploy the final model to a production environment for real-time forecasting. Monitor the model's performance and update it as necessary with new data.

4.2 Data Inputs

Plant Generation Dataset:

Weather Sensor dataset:

4.3 Time based Train Test Split:

4.4 Model Building:

4.4.1 Model Algorithm:

Steps 1: Split Data into k-Folds:

• Shuffle and split df_train into k equal-sized folds.

• Initialize an empty list “rmse_scores” to store RMSE scores for each model.

Step 2: Initialize an empty list “model_rmse_scores” to store RMSE scores for each iteration of k-fold cross-validation.

• For each fold from 1 to k - Set the current fold as the validation set. Combine the remaining k-1 folds to form the training set.

• Split Data: xtrain and ytrain are used for variable for training. xvalid and yvalid are for validation.

• Standardize Data: After fitting a Standard Scaler on xtrain and transforming both xtrain and xvalid.

• Train the Model: Fit the model on the standardized xtrain and ytrain.

• Predict and Evaluate: Predict yvalid using the trained model. Calculate the RMSE between predicted and actual yvalid.

Step 3:Calculate Mean RMSE: Compute the mean RMSE for the model from model_rmse_scores. Append the mean RMSE to rmse_scores.

Step 4: Compare Result: Compare the mean RMSE scores for each model to identify the best performing model.

MATCHED SOURCES:

Is it possible to train a deep learning model using only tabular ...

https://aidaily.quora.com/Is-it-possible-to-train-a-deep-lea. .....(https://aidaily.quora.com/Is-it-possible-to-train-a-deep-learning-model-using-only-tabular-data-no-images-If-yes-how?top_ans=1477743703902565)

www.linkedin.com › company › datathickDataThick | LinkedIn

https://www.linkedin.com/company/datathick/ (https://www.linkedin.com/company/datathick/)

Report Generated on **June 12, 2024** by https://www.check-plagiarism.com/ (https://www.check-plagiarism.com/)

PLAGIARISM SCAN REPORT

Date	June 12, 2024
Exclude URL:	NO

Unique Content	**95**	Word Count	914
Plagiarized Content	**5**	Records Found	0

CONTENT CHECKED FOR PLAGIARISM:

4.4.2 Algorithm Selection:

Linear Regression: A simple yet effective method for predicting continuous variables, suitable for modeling the relationship between solar irradiation and power generation.

Decision Trees: Can capture non-linear relationships and interactions between variables.

Random Forests: An ensemble method that improves prediction accuracy by averaging multiple decision trees.

Support Vector Machines (SVM): Effective for regression tasks with high-dimensional data.

Neural Networks: Can model complex relationships and interactions between features, useful for capturing non-linear dependencies in the data.

4.4.3 Choosing RMSE over MAE and MSE:

Metric Formula Description Use Case

MAE 1/n ∑_(i=1)^n▒|y_i^real 〖-y〗 _i^pred | Measures the average magnitude of errors in a set of predictions, without considering their direction Used to evaluate the accuracy of regression models

MSE 1/n ∑_(i=1)^n▒ 〖 〖(y〗 _i^real-y_i^pred)〗 ^2 Average of squared errors between actual and predicted values Penalizes larger errors more than MAE

RMSE √1/n ∑_(i=1)^n▒ 〖 〖(y〗 _i^real-y_i^pred)〗 ^2 Square root of the average of squared errors Further amplifies the impact of larger errors, preferred for this problem

4.5 Model Training

Using Scikit-learn Pipelines: Utilizing Scikit-learn's pipeline functionality to streamline the training process. This allows for easy management and comparison of multiple models.

Regression Algorithms: Training seven different regression algorithms with default parameters, including a 3-

layered Neural Network regressor.

Standardization: Before training, we standardize the features to ensure that all variables are on the same scale.

4.6 Hyperparameter Tuning using Randomized Search

Hyperparameters are settings that need to be tuned to achieve the best possible performance from a model. For Random Forest Regressor, we focused on optimizing the following hyperparameters:

Number of Trees (n_estimators): Represents the number of trees in the forest.

Maximum Depth of Trees (max_depth): Controls the maximum depth of each tree.

Minimum Samples Split (min_samples_split): Defines the minimum number of samples required to split an internal node. Higher values prevent the model from learning overly specific patterns.

Minimum Samples Leaf (min_samples_leaf): The minimum number of samples required to be at a leaf node.

Maximum Features (max_features): Specifies the number of features to consider when looking for the best split.

Criterion: MSE (Mean Squared Error): Measures the average of the squares of the errors or deviations, which is useful for regression tasks.

Chapter 5

Results

5.1 Model Performance Comparison

Regressor RMSE (kW)

Linear Regression 2.3738

Decision Tree Regressor 2.3599

Random Forest Regressor 1.7387

Ridge Regressor 2.3774

Lasso Regressor 2.7925

XG Boost Regressor 1.8680

ANN Regressor 3.3768

Random Forest Regressor has performed the best so we are going to predict the data with Random Forest Regressor.

5.2 Predictions

After taking last 3 day's data for testing and removing all the data leakage we have predicted the RMSE for test data.

Input:

Output:

METRICS VALUE (kW)

Test RMSE 1.86830155234851

Though, the RMSE on Test set is slightly higher than what we had for Training data(1.7386 kW) but seems good considering the less amount of training data.

5.3 Actual vs Predicted data plot

Chapter 6

Uses & Applications

6.1 Applications

Energy Production Optimization: Accurate forecasting helps in optimizing the operation of solar power plants by predicting power generation and aligning it with demand.

Grid Stability: Reliable solar power forecasts assist grid operators in maintaining stability by effectively integrating solar power with other energy sources.

Cost Savings: Better forecasts can reduce the need for backup power and lower operational costs, leading to economic benefits for both utilities and consumers.

Maintenance Planning: Predictive insights can inform maintenance schedules, ensuring solar panels and equipment are serviced during low-production periods, thus maximizing uptime.

Policy and Planning: Governments and energy planners can use forecasting data to develop and implement policies for sustainable energy growth and infrastructure development.

Market Trading: Energy traders can leverage accurate forecasts to make informed decisions in energy markets, optimizing financial returns.

Consumer Awareness: Providing consumers with forecast data can promote energy-saving practices by aligning energy consumption with peak solar production times.

6.2 Future Scope

Integrating of IoT devices can enhance the accuracy, efficiency, and real-time capabilities of our forecasting model.

IoT enables remote management, anomaly detection, and predictive maintenance of solar systems.

High-frequency data from IoT devices improves the model’s ability to capture short-term fluctuations.

IoT-assisted forecasts help balance energy supply and demand dynamically within smart grids.

6.3 Limitations

Data Quality and Availability: The accuracy of forecasts is highly dependent on the quality and completeness of historical and real-time data. Missing or inaccurate data can significantly impact model performance.

Weather Dependency: Solar power generation is heavily influenced by weather conditions, which can be unpredictable and variable. Sudden changes in weather can lead to forecasting errors.

Overfitting: Complex models might overfit to historical data, performing well on past data but poorly on new, unseen data. Proper cross-validation and regular model updates are necessary to mitigate this risk.

External Factors: Factors such as dust, shading, and equipment malfunctions are not always predictable and can affect the accuracy of solar power forecast.

Chapter 7

References

7.1 References

7.1.1 Books

[1] ”Solar Energy: The Physics and Engineering of Photovoltaic Conversion, Technologies and Systems"

Klaus Jäger, Olindo Isabella, Arno Smets, Rene van Swaaij, Miro Zeman

[2] ”Machine Learning and Data Science in the Power Generation Industry: Best Practices, Tools, and Case Studies"

Patrick Bangert

[3] ”Data Science for Supply Chain Forecasting"

Nicolas Vandeput

7.1.2 Journals

[1] ”A Comprehensive Review on Solar Power Forecasting"

U. A. Amam et al. , Renewable and Sustainable Energy Reviews, Journal of Machine Learning, volume 1, 2019

[2] “Data Preprocessing for Machine Learning in Solar Energy Forecasting"

T. Chen et al. , Renewable and Sustainable Energy Reviews, Journal of Machine Learning, volume 1, 2019

7.1.3 Datasets

https://www.kaggle.com/datasets/anikannal/solar-power-generation-data

MATCHED SOURCES:

medium.com › @ambika199820 › understanding-decisionUnderstanding Decision Trees. “A decision tree is ... - Medium

https://medium.com/@ambika199820/understanding-decision-tree. ....(https://medium.com/@ambika199820/understanding-decision-trees-df9455f02581/)

chrisyandata.medium.com › understanding-decisionUnderstanding Decision Trees: Structure, Splitting Nodes ...

https://chrisyandata.medium.com/understanding-decision-trees. ..... (https://chrisyandata.medium.com/understanding-decision-trees-structure-splitting-nodes-parameters-and-example-63af1c72b59d/)

Solar Energy The Physics and Engineering of Photovoltaic ...

https://www.chegg.com/textbooks/solar-energy-1906860327 (https://www.chegg.com/textbooks/solar-energy-1906860327)

Best Practices, Tools, and Case Studies (Paperback)

https://www.walmart.com/ip/Machine-Learning-and-Data-Science. .....(https://www.walmart.com/ip/Machine-Learning-and-Data-Science-in-the-Power-Generation-Industry-Best-Practices-Tools-and-Case-Studies-Paperback-9780128197424/862015973)