

Predict The Electrical Power Out From Combined Cycle Power Plant Using IBM Watson.

CONTENTS

SNO	TOPIC	PAGE NO
1	INTERODUCTION and PURPOSE	2
2	ARCHITECTURE	3
3	SOFTWARE INSTALLATIONS AND DATA SET	4
4	READING DATA SETS	5
5	EXPLORATORY DATA ANALYSIS	5,6
6	CHECKING FOR NULL VALUES	6,7
7	SPLIT THE DEPENDENT AND INDEPENDENT FEATURES INTO TRAIN SET AND TEST SET	7,8
8	Model building	9
9	Model prediction	9,10
10	Application building	10
11	CONCLUSION	11,12

INTRODUCTION

OVERVIEW:

Project Title: Predict The Electrical Power Out From Combined Cycle Power Plant Using IBM Watson

A brief description about our project:

- a. The Combined Cycle power plant or combined cycle gas turbine, a gas turbine generator generates electricity and waste heat is used to make steam to generate additional electricity via a steam turbine.
- b. The gas turbine one of the most efficient one for the conversion of gas fuels to mechanical power or electricity. Combined cycle power plants are frequently used for power production.
- c. These days prediction of power plant output based on operating parameters is a major concern and power plant influenced by four main parameters.
- d. Which are used as input variables in the data set, such as ambient temperature, atmospheric pressure, relative humidity and exhaust steam pressure.

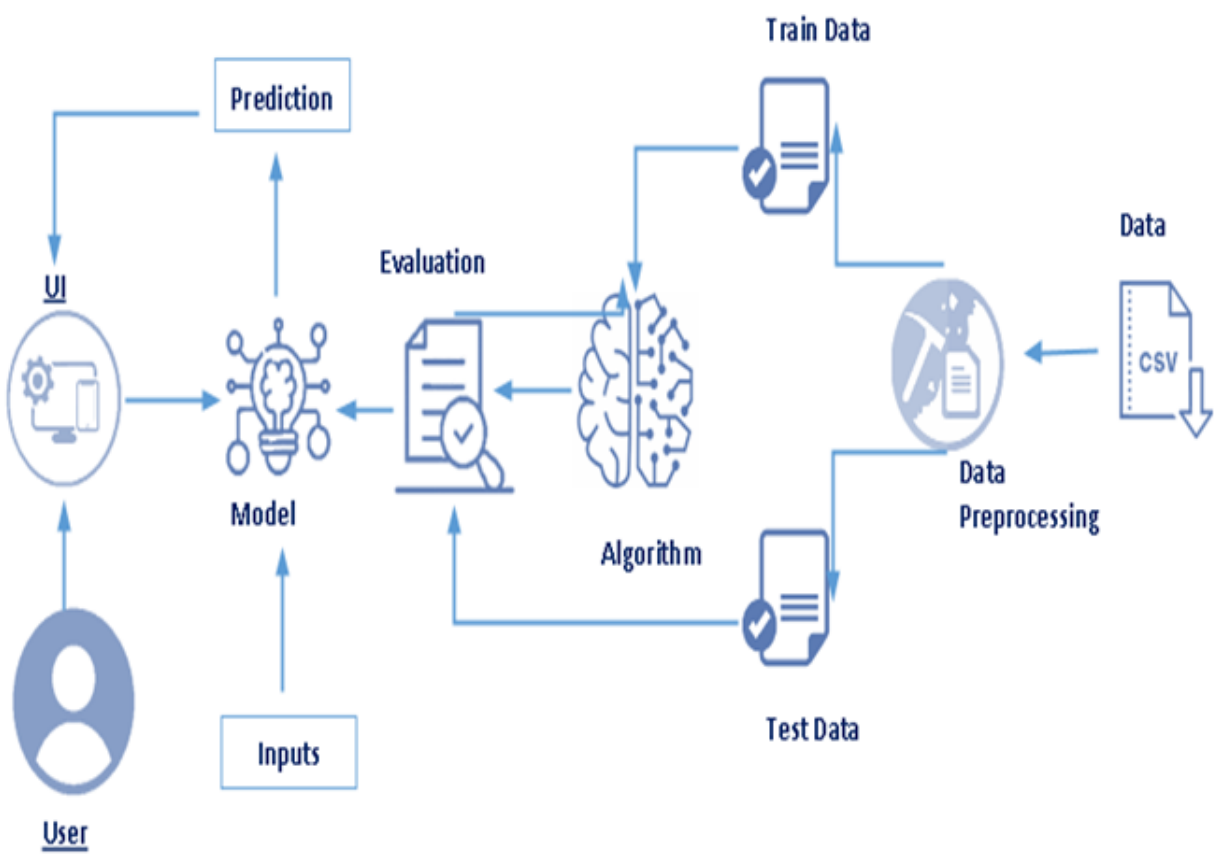
PURPOSE:

1. The use of this project what we can archived using this.
2. Predict The Electrical Power Out From Combined Cycle Power Plant

dashboard we well

3. know fundamental concepts and can work on Anaconda.
4. gain a board understanding of plotting different graphs.
5. Able to create a meaningful dash boards.
6. can make any data sets to understand different graphs.

Architecture:



Software Installation:

Anaconda Navigator:

1. Anaconda Navigator is a free and open-source distribution of the Python and R programming languages for data science and machine learning related applications.
2. It can be installed on Windows, Linux, and Mac OS.
3. Conda is an open-source, cross-platform, package management system.
4. Anaconda comes with so very nice tools like JupyterLab, Jupyter Notebook,
5. QtConsole, Spyder, Glueviz, Orange, Rstudio, Visual Studio Code. For this project, we will be using Jupyter notebook and Spyder.

Data Set:

For this article we will be using the dataset provided by IBM which is available at the UCI Machine Learning repository we are using 00294 data set.

<https://archive.ics.uci.edu/ml/machine-learning-databases/00294/>

Importing Libraries :

The first step is usually importing the libraries that will be needed in the program.

The required libraries to be imported to Python script are:

- Numpy: It is an open-source numerical Python library. It contains a multi-dimensional array and matrix data structures.
- It can be used to perform mathematical operations on arrays such as trigonometric, statistical, and algebraic routines.
- Pandas: It is a fast, powerful, flexible and easy to use open source data analysis and manipulation tool, built on top of the Python programming language.
- Matplotlib: Visualisation with python. It is a comprehensive library for creating static, animated, and interactive visualizations in Python.
- Seaborn: Seaborn is a library for making statistical graphics in Python.
- Seaborn helps you explore and understand your data.
- Its plotting functions operate on data frames and arrays containing whole datasets and internally perform the necessary semantic mapping and statistical aggregation to produce informative plot.
- Pickle: The pickle module implements serialization protocol, which provides an ability to save and later load Python objects using special binary format.

Reading the Dataset:

1. You might have your data in .csv files, .excel files or **.tsv** files or

something else.

2. first step will be to read it into a data structure that's compatible with panda.
3. Let's load a .csv data file into pandas. There is a function for it, called **read_csv()**.
4. Names on Windows tend to have backslashes in them. But we want them to mean actual backslashes, not special characters.
5. Temperature (T) in the range 1.81°C and 37.11°C.
6. Ambient Pressure (AP) in the range 992.89-1033.30 milibar.
7. Relative Humidity (RH) in the range 25.56% to 100.16%
8. Exhaust Vacuum (V) in teh range 25.36-81.56 cm Hg.
9. The averages are taken from various sensors located around the plant that record the ambient variables every second. The variables are given without normalization.
- 10.

Exploratory Data Analysis :

head() :To check the first five rows of the dataset, we have a function called **head()**.

Tail(): To check the last five rows of the dataset, we have a function called **tail()**

Understanding Data Type and Summary of features:

How the information is stored in a Data Frame or Python object affects what we can do with it and the outputs of calculations as well.

There are two main types of data those are numeric and text data types.

- Numeric data types include integers and floats.
- Text data type is known as Strings in Python, or Objects in Pandas. Strings can contain numbers and / or characters.
- or example, a string might be a word, a sentence, or several sentences.

Checking For Null Values:

1. After loading it is important to check the complete information of data as it can indicate many of the hidden information such as null values in a column or a row

2. Check whether any null values are there or not. if it is present then following can be done,

- a. Imputing data using Imputation method in sklearn
- b. Filling NaN values with mean, median and mode using fillna() method.

Data Visualization:

- Data visualization is where a given data set is presented in a graphical format. It helps the detection of patterns, trends and correlations that might go undetected in text-based data.
- To visualize the dataset we need libraries called Matplotlib and Seaborn.
- The Matplotlib library is a Python 2D plotting library which allows you to generate plots, scatter plots, histograms, bar charts etc..

Splitting The Dataset Into Dependent And Independent Variable

- In machine learning, the concept of dependent variable (y) and

independent variables(x) is important to understand.

- Here, Dependent variable is nothing but output in dataset and independent variable is all inputs in the dataset.
- With this in mind, we need to split our dataset into the matrix of independent variables and the vector or dependent variable.
- Mathematically, Vector is defined as a matrix that has just one column.
- To read the columns, we will use `iloc` of pandas (used to fix the indexes for selection) which takes two parameters — [row selection, column selection].
- list of the independent variable x with our selected columns and for dependent variable y we are only taking the PE column.

Split The Dependent And Independent Features Into Train Set And Test Set

- When you are working on a model and you want to train it, you obviously have a dataset. But after training, we have to test the model on some test dataset.
- For this, you will a dataset which is different from the training set you used earlier.
- But it might not always be possible to have so much data during the development phase
- In such cases, the solution is to split the dataset into two sets, one for training and the other for testing.
- But the question is, how do you split the data? You can't possibly manually split the dataset into two sets.
- And you also have to make sure you split the data in a random manner.
- To help us with this task, the Scikit library provides a tool, called the Model Selection library.
- There is a class in the library which is, '[train_test_split](#).' Using this we can easily split the dataset into the training and the testing datasets in various proportions.
- The train-test split is a technique for evaluating the performance of a machine learning algorithm.

- Train Dataset: Used to fit the machine learning model.
- Test Dataset: Used to evaluate the fit machine learning model.
- In general you can allocate 80% of the dataset to training set and the remaining 20% to test set.
- We will create 4 sets— `x_train` (training part of the matrix of features), `x_test` (test part of the matrix of features), `y_train` (training part of the dependent variables associated with the X train sets, and therefore also the same indices), `y_test` (test part of the dependent variables associated with the X test sets, and therefore also the same indices).
- There are a few other parameters that we need to understand before we use the class:
- `test_size` — this parameter decides the size of the data that has to be split as the test dataset. This is given as a fraction.
- For example, if you pass 0.5 as the value, the dataset will be split 50% as the test dataset.
- `train_size` — you have to specify this parameter only if you're not specifying the `test_size`.
- This is the same as `test_size`, but instead you tell the class what percent of the dataset you want to split as the training set.
- `random_state` — here you pass an integer, which will act as the seed for the random number generator during the split.
- Or, you can also pass an instance of the `Random_state` class, which will become the number generator.
- If you don't pass anything, the `Random_state` instance used by `np.random` will be used instead.
- Now split our dataset into train set and test using `train_test_split` class from scikit learn library.

Model Building:

Predictive modeling is a mathematical approach to create a statistical model to forecast future behavior based on input test data.

Train Model:

- After assigning the algorithm and getting the data handy, we train our

model using the input data applying the preferred algorithm.

- It is an action to determine the correspondence between independent variables, and the prediction targets.

Model Prediction:

We make predictions by giving the input test data to the trained model. We measure the accuracy by using a cross-validation strategy or ROC curve which performs well to derive model output for test data.

Model building includes the following main tasks

1. Training and testing the model
2. Evaluation of Model
3. Save the model
4. Predicting the output using the model

-

-

Train and Test the Model :

There are several Machine learning algorithms to be used depending on the data you are going to process such as images, sound, text, and numerical values.

The algorithms that you can choose according to the objective that you might have may be Classification algorithms are Regression algorithms.

-

Model Evaluation:

Finally, we need to check to see how well our model is performing on the test data.

Save the Model:

After building the model we have to save the model.

Pickle is used for serializing and de-serializing Python object structures, also called marshalling or flattening.

Later on, this character stream can then be retrieved and de-serialized back

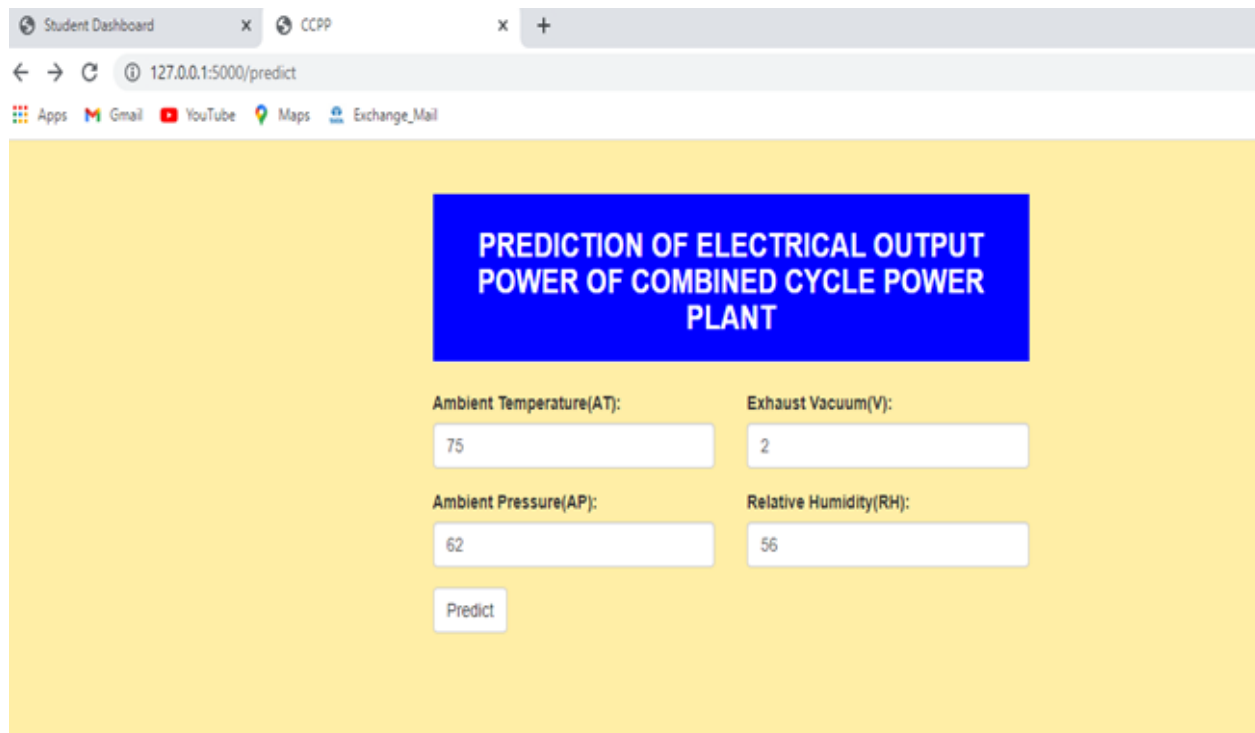
to a Python object.

Application Building:

- a. Create HTML file
- b. Build Python Code
- c. Run the app

Predicting the output using the model

Giving Inputs:

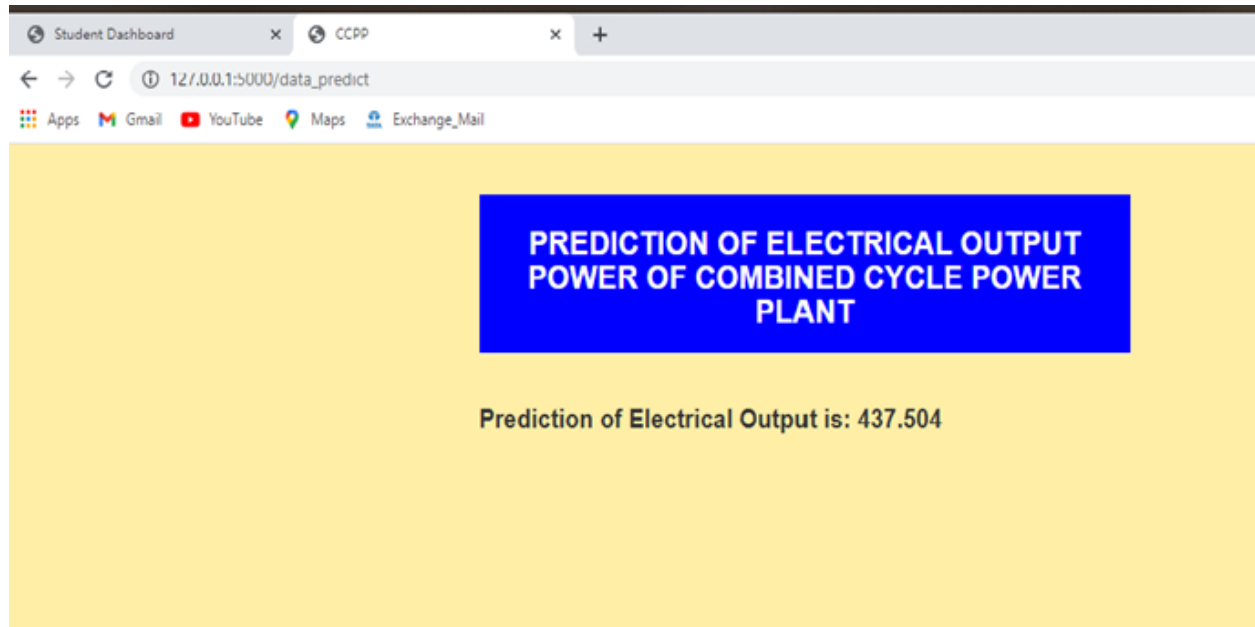


The screenshot shows a web browser window with two tabs: 'Student Dashboard' and 'CCPP'. The address bar displays '127.0.0.1:5000/predict'. Below the browser window, a web application interface is visible on a yellow background. At the top, a blue box contains the text 'PREDICTION OF ELECTRICAL OUTPUT POWER OF COMBINED CYCLE POWER PLANT'. Below this, there are four input fields arranged in a 2x2 grid. The first row contains 'Ambient Temperature(AT):' with a value of '75' and 'Exhaust Vacuum(V):' with a value of '2'. The second row contains 'Ambient Pressure(AP):' with a value of '62' and 'Relative Humidity(RH):' with a value of '56'. A 'Predict' button is located at the bottom left of the input area.

Input Parameter	Value
Ambient Temperature(AT)	75
Exhaust Vacuum(V)	2
Ambient Pressure(AP)	62
Relative Humidity(RH)	56

Predict

Predicting Output for given inputs:



CONCLUSION:

1. At the beginning of this article, we set out to develop a predictive model for full-load output power (PE) based on the dataset provided.
2. We explored the dataset to find out if we had missing values or other problems, then played around with 4 features subset selections on 3 different machine learning regression algorithms.
3. We were able to discover that using a complete set of parameters or features on the Random Forest Regression algorithm yielded the best results.

