**Predicting the Energy Output of Wind Turbine based on Weather Conditions**

**1. INTRODUCTION**

1.1 Overview

Wind energy plays an increasing role in the supply of energy worldwide. The energy output of a wind farm is highly dependent on the weather conditions present at its site. If the output can be predicted more accurately, energy suppliers can coordinate the collaborative production of different energy sources more efficiently to avoid costly overproduction. In this project, we predict energy prediction based on weather data and analyze the important parameters as well as their correlation on the energy output.

1.2 Purpose

The main purpose is to map weather data for energy production. We wish to show that even data that is publicly available for weather stations close to wind farms can be used to give a good prediction of the energy output. Furthermore, we examine the impact of different weather conditions on the energy output of wind farms.

**2. LITERATURE SURVEY**

2.1 Existing problem

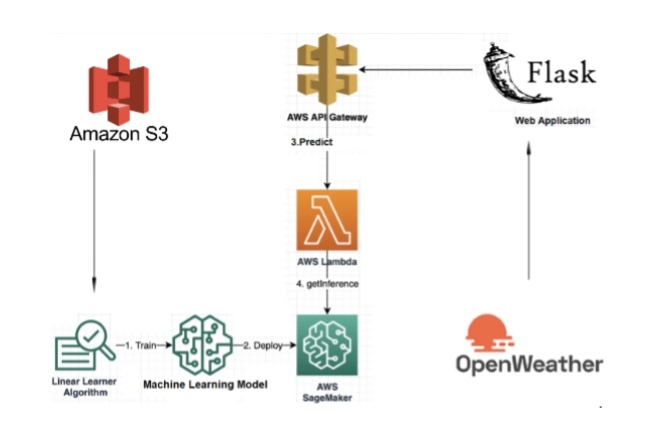
There exist a number of technological, environmental and political challenges linked to supplementing existing electricity generation capacities with wind energy. Here, mathematicians and statisticians could make a substantial contribution at the interface of meteorology and decision-making, in connection with the generation of forecasts tailored to the various operational decision problems involved. Indeed, while wind energy may be seen as an environmentally friendly source of energy, full benefits from its usage can only be obtained if one is able to accommodate its variability and limited predictability. Based on a short presentation of its physical basics, the importance of considering wind power generation as a stochastic process is motivated. The conventional moving-average statistical models were proven to be less efficient in forecasting the wind energy, as the wind speed is inherently variable quantity.

2.2 Proposed solution

To overcome the disadvantages of conventional models , advanced ensemble models such as XGBoost , can be used to map the inherently variable attribute to a complex function.It is an optimized distributed gradient boosting library designed to be highly efficient , flexible and portable. . It implements machine learning algorithms under the Gradient Boosting framework. XGBoost provides a parallel tree boosting (also known as GBDT, GBM) that solve many data science problems in a fast and accurate way. The same code runs on major distributed environment (Hadoop, SGE, MPI) and can solve problems beyond billions of examples.

**3. Thoeritical Analysis**

3.1 Block diagram



3.2 Hardware / Software designing

Amazon **SageMaker** is a fully managed service that provides every developer and data scientist with the ability to build, train, and deploy machine learning (ML) models quickly. **SageMaker** removes the heavy lifting from each step of the machine learning process to make it easier to develop high quality models.The steps to be followed are

1. Training SageMaker Model

2. Deploying SageMaker Model

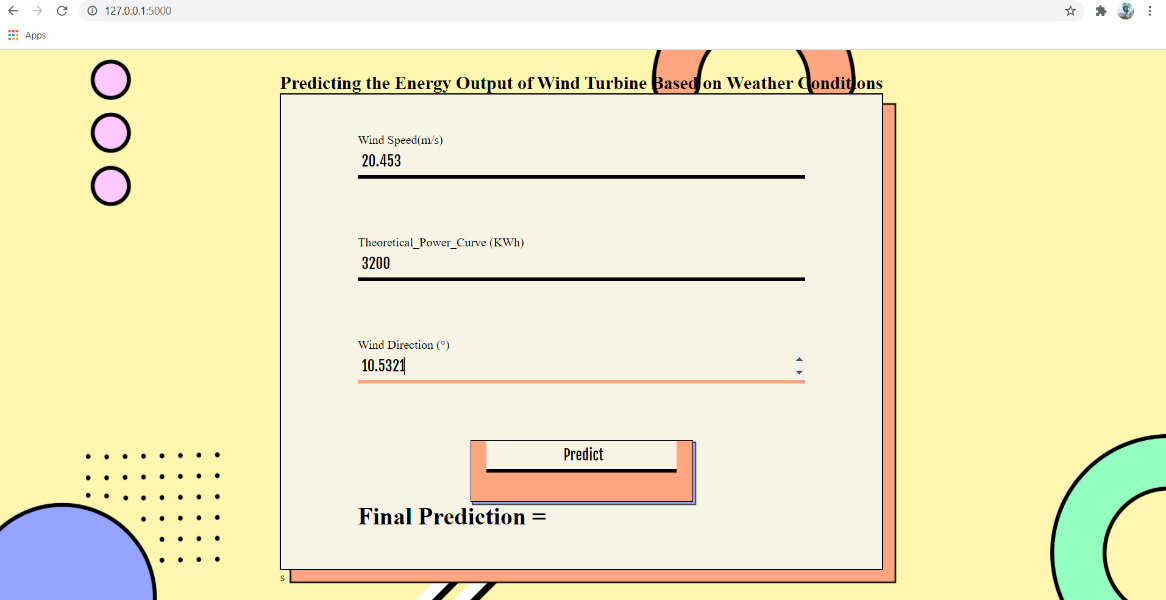
3. Creating Lambda

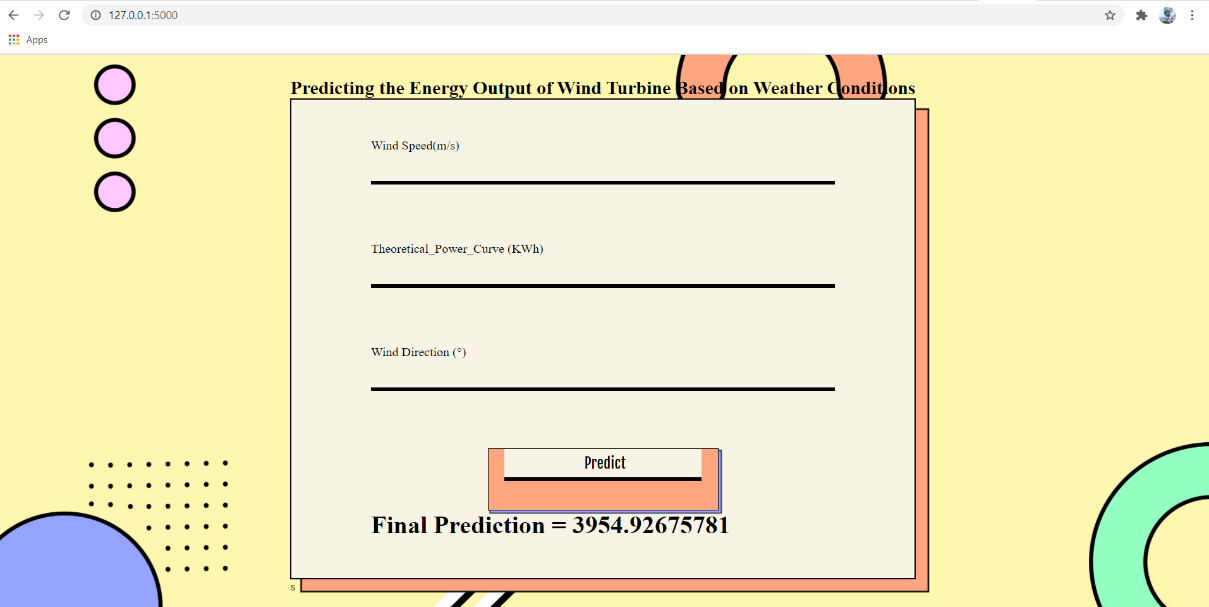
4. Creating API Gateway

5. Building Flask application

**4. Experimental Investigation**

Experimentally the model showed good accuracy and sample results where as follows:





**5. Flowchart**

****

**6.Result**

1.We report on the correlation of the different variables for the energy output.

2.Model gives a very reliable prediction of the energy output.

3.We predict the energy output with accuracy up to 80%.

**7.Advantages and Disadvantages**

**Advanages**

1.Easy to use and has a friendly user interface to work with.

2.Reduces man power and cost efficient

3.Faster Predictions on just a click of a button

4.Intuitive User Interface

**Disadvanages**

1.Requires all services that handles requests and renders responses.

2.Requires some complex integration of services

**8. Applications**

1.Several companies can use the service and deploy it,on their own servers.

2.This would save time and money as no three to four applications are needed.

3.The service can be provided to users in application along with other features.

4.Could be helpful even in areas with less connectivity.

5. As the application is quite robust and resilient in its architecture,it allows one to easily navigate through different sections.

**9. Conclusion**

A Machine Learning has been developed to forecast the Power output of the wind turbine. A simple, efficient and a versatile model is built keeping in mind the diversity of data, computational complexity and overhead involved in making API calls to the model for prediction. The product can increase the accuracy of the forecasting the output from the wind turbine .Overall accurate wind power prediction reduces the financial and technical risk of uncertainty of wind power production for all electricity market participants.

**10. Future Scope**

1. A more generalized model can be developed to suit forecasting for different locations.

2. The model can be developed to make predictions on other sources of data such as solar power, tidal power etc.

3. On-device model can be developed to make much more faster predictions.

**11. BIBILOGRAPHY**

**1. SageMaker -** <https://aws.amazon.com/getting-started/hands-on/build-train-deploy-machine-learning-model-sagemaker/>

**2. Lamda -** <https://docs.aws.amazon.com/lambda/latest/dg/getting-started-create-function.html>

3.API Gateway - <https://docs.aws.amazon.com/lambda/latest/dg/getting-started-create-function.html>

4. S3 - <https://docs.aws.amazon.com/AmazonS3/latest/dev/Introduction.html>

5. Vladislavleva, Ekaterina (Katya) & Friedrich, Tobias & Neumann, Frank & Wagner, Markus. (2013). Predicting the Energy Output of Wind Farms Based on Weather Data: Important Variables and their Correlation. Renewable Energy. 50. 236-243. 10.1016/j.renene.2012.06.036.

6. <https://doi.org/10.1016/j.renene.2012.06.036>

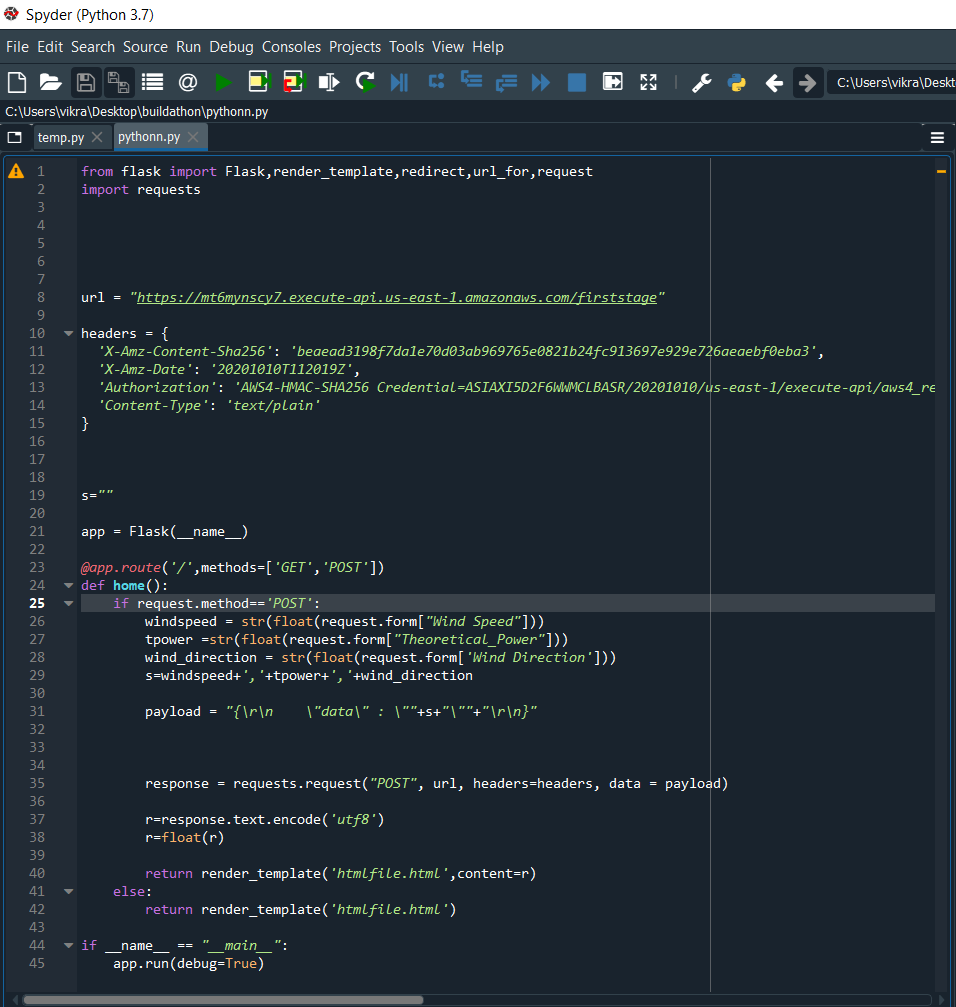
**APPENDIX**

A. Source code

**Onedrive drive:**

https://1drv.ms/u/s!Agf8vAmtMZLcecMkSEwBdYjEpD0?e=Ji1UsP

Flask Code



**SageMaker Code:**

<https://projectmlnotebook.notebook.us-east-1.sagemaker.aws/nbconvert/html/Energyprediction.ipynb?download=false>

**Lamda Code**

import os

import io

import boto3

import json

import csv

# grab environment variables

ENDPOINT\_NAME = "xgboost-2020-10-07-16-33-01-743"

runtime= boto3.client('runtime.sagemaker')

def lambda\_handler(event, context):

print("Received event: " + json.dumps(event, indent=2))

data = json.loads(json.dumps(event))

payload = data['data']

print(payload)

response = runtime.invoke\_endpoint(EndpointName=ENDPOINT\_NAME,

ContentType='text/csv',

Body=payload)

print(response)

result = json.loads(response['Body'].read().decode())

print(result)

return (result)