

IBM-Recommendation-Model- To-Utilize-Energy-From-Wind- Farm

- *By Horoscopers*

SCOPE

- Our ultimate aim is to competitive real time, reduce energy wastage and losses in power grid.
- By using our model, we can predict the valuable time of power production.
- Rest of plant can be provided frequently by predicting the unworthy time.
- By forecasting energy, we can solve the future energy needs.
- As well availability of energy can be determined.

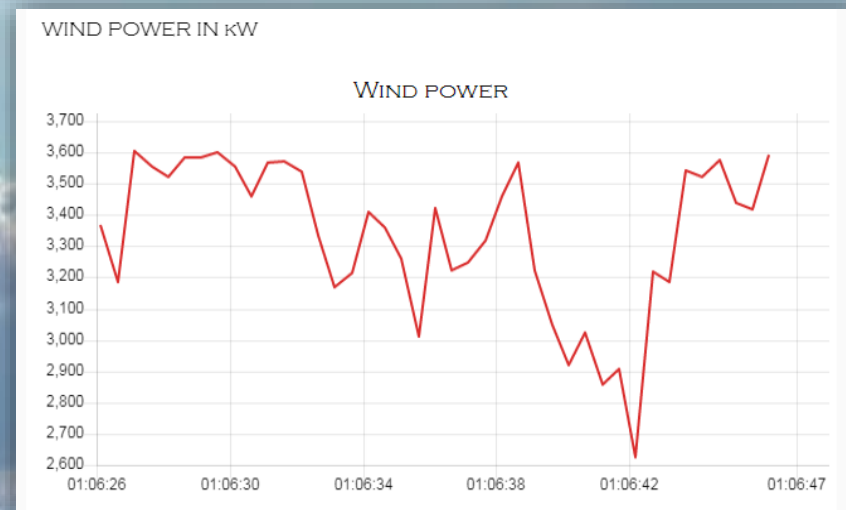
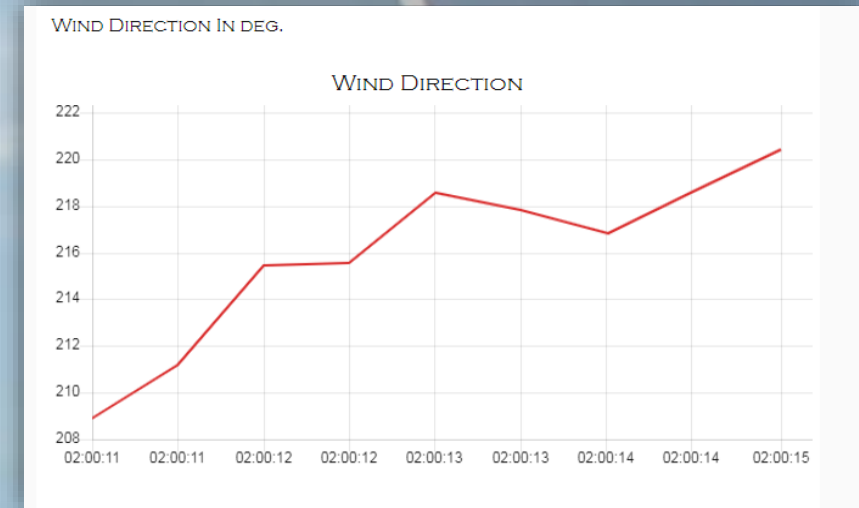
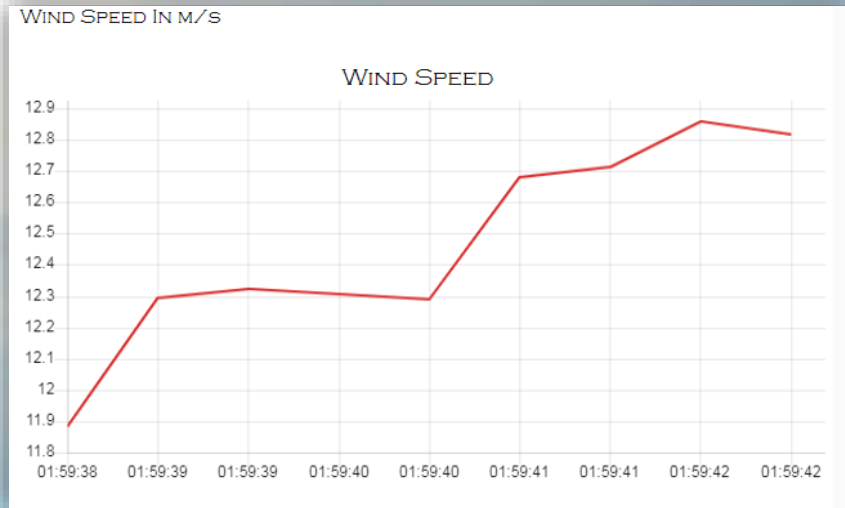
LITERATURE REVIEW:-

Article A New Hybrid Approach to Forecast Wind Power for Large Scale Wind Turbine Data Using Deep Learning with TensorFlow Framework and Principal Component Analysis.

- Mansoor Khan , Tianqi Liu ,* and Farhan Ullah

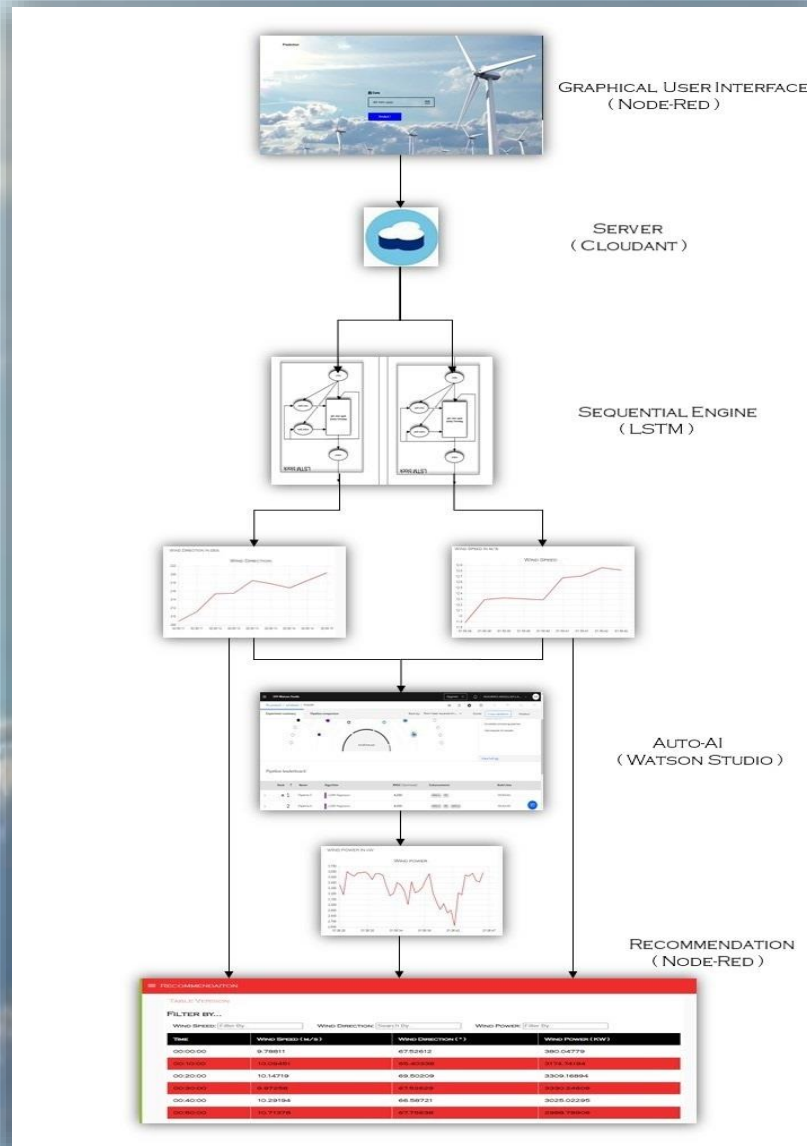
- The principal components are then used as input data to **deep learning** based on the **TensorFlow framework**. PCA provides feature extraction and selection.
- Here, acc, val_loss, val_acc represents **accuracy, validation loss** and **validation accuracy**, respectively. The loss, accuracy and loss, and validation loss are calculated for wind power forecasting.
- The proposed deep learning algorithm is applied to PCs to **forecast wind power**. The **Keras** API is used with TensorFlow to configure a more reliable **neural network**.

Forecasting output



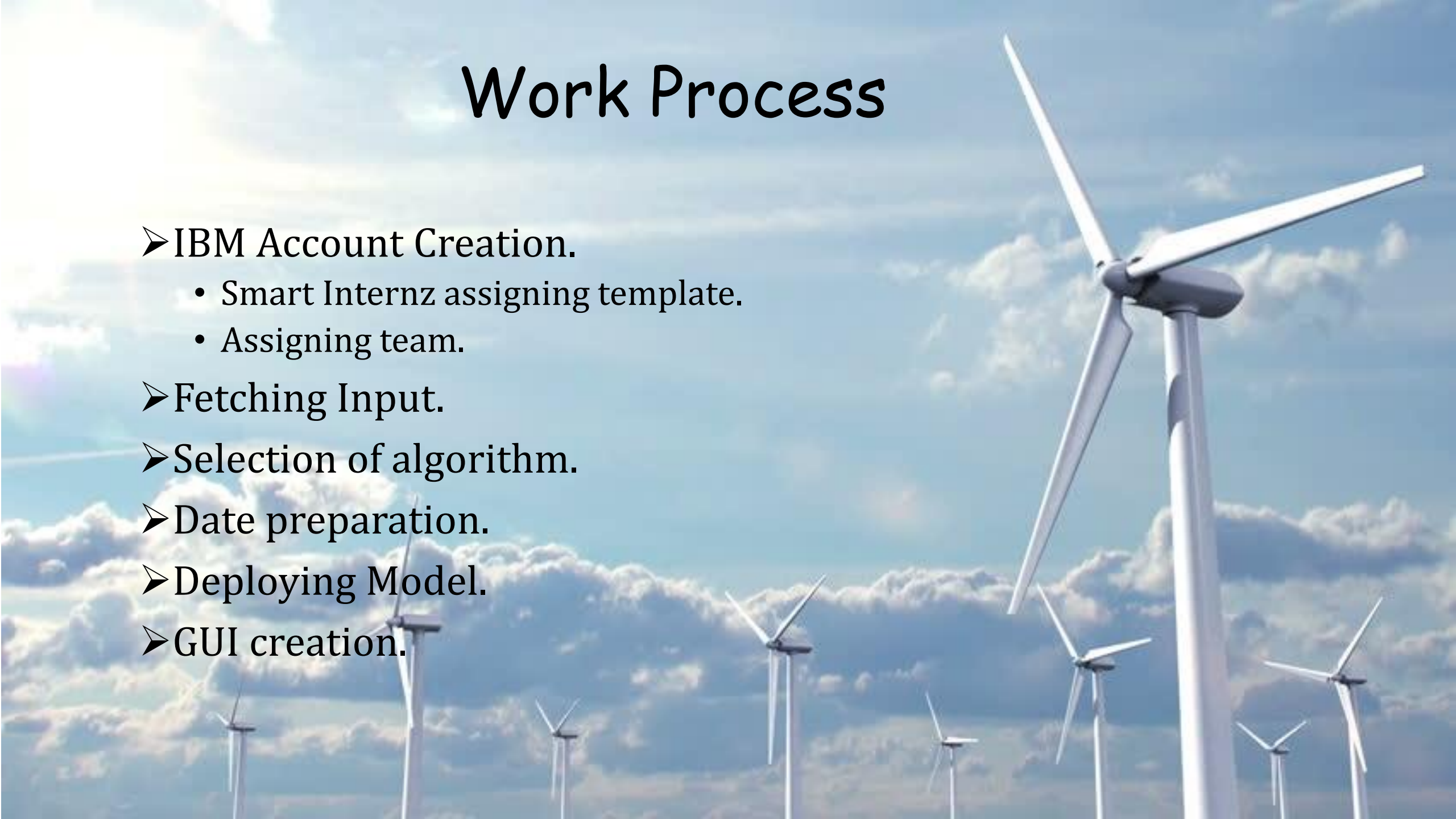
The Above Forecasting Done For 24 Hrs On Date 28-10-2021

Architectural Flow



Work Process

- IBM Account Creation.
 - Smart Internz assigning template.
 - Assigning team.
- Fetching Input.
- Selection of algorithm.
- Data preparation.
- Deploying Model.
- GUI creation.



Fetching Input:-

- Collecting of data from various website.
 - Kaggle-<https://www.kaggle.com/berkerisen/wind-turbine-scada-dataset>
 - Github-https://github.com/ShashwatArghode/Wind-Energy-Prediction-using-LSTM/blob/master/AL_WIND_07_12.xlsx
- Importing data in Watson Studio in a click.

```
In [10]: import types
import pandas as pd
from botocore.client import Config
import ibm_boto3

def __iter__(self): return 0

# @hidden_cell
# The following code accesses a file in your IBM Cloud Object Storage. It includes your credentials.
# You might want to remove those credentials before you share the notebook.
client_64c85547f35d43b5b41b68c871da81d6 = ibm_boto3.client(service_name='s3',
    ibm_api_key_id='HcWLNsvs1l76yIZ0P9bQ1EHbvJ5Ty-JY0o0ShzpA1Fll',
    ibm_auth_endpoint='https://iam.cloud.ibm.com/oidc/token',
    config=Config(signature_version='oauth'),
    endpoint_url='https://s3.eu-geo.objectstorage.service.networklayer.com')

body = client_64c85547f35d43b5b41b68c871da81d6.get_object(Bucket='hanif-donotdelete-pr-nxileyclobvpqp',Key='winddirtime.csv')['Body']
# add missing __iter__ method, so pandas accepts body as file-like object
if not hasattr(body, "__iter__"): body.__iter__ = types.MethodType( __iter__, body )

series = pd.read_csv(body,index_col='Date/Time')
series.head()
```

Out[10]:

	Wind Direction (°)
Date/Time	
01 01 2018 00:00	259.994904
01 01 2018 00:10	268.641113
01 01 2018 00:20	272.564789
01 01 2018 00:30	271.258087
01 01 2018 00:40	265.674286

Selection Of Algorithm

- Choosing Best algo . is the main process ever in forecasting.
- Normally algo. Like **SVM, Arima, Auto arima** are used for time series prediction .
- But in this recommendation model we tried **LSTM algo** ,which is one of library of **keras**.
- We have done this forecasting in **three segment**, Wind Speed , Direction, Power.
- Wind Speed and Wind Direction using **lstm** model.
- Power Forecasting using **AutoAi**, The facility provided by our **IBM Watson studio**.

Feature Engineering

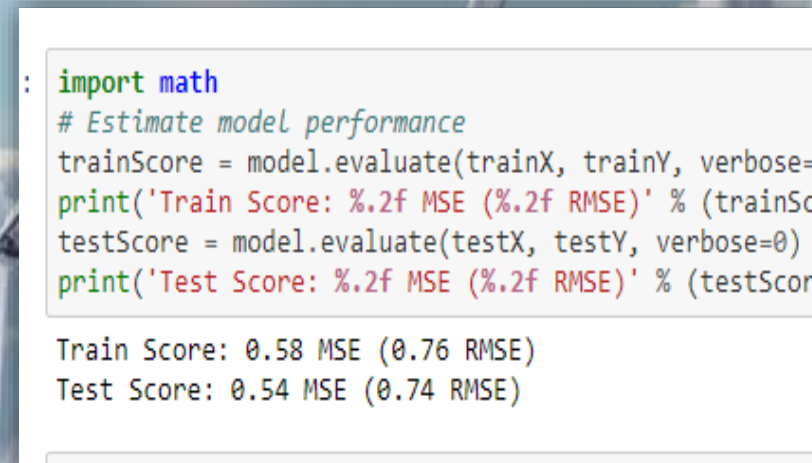
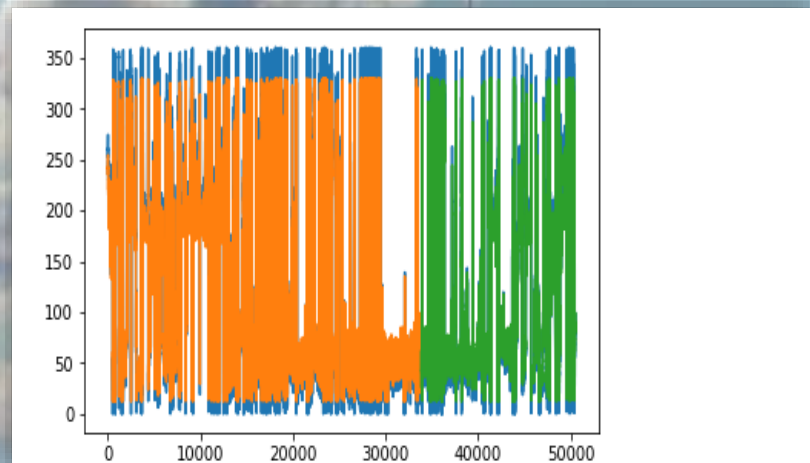
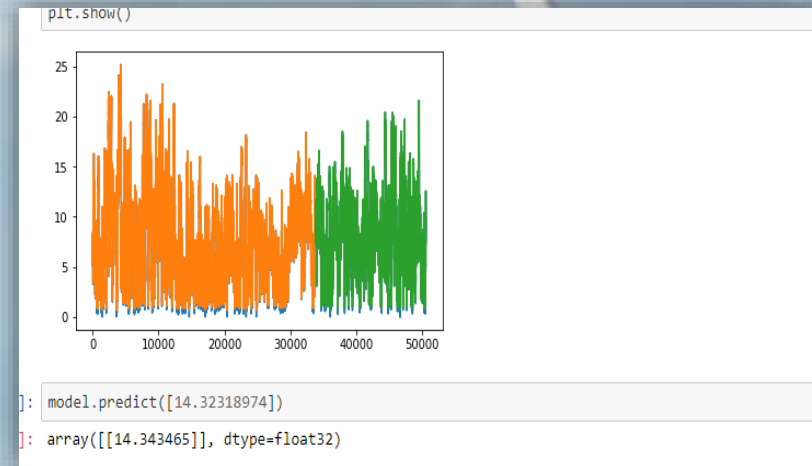
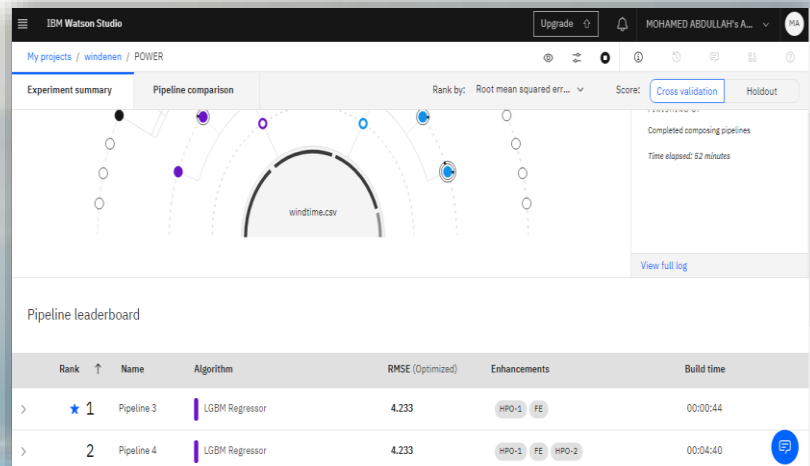
FOR WIND SPEED AND DIRECTION:-

- To fit our model we done some **preprocessing** .
- Split up data into **train** and **test** .
- Converting the data into scalar form , then take difference of past data.
- Setting the epoch ,finding study rate, fix the batch size, selection of neurons and validate.

FOR POWER:-

- We done it in **auto AI**.
- For power prediction we give the **input from the output** of former prediction.

Validation of our Prediction



Deployment

- IBM Watson Studio made Deployment easy in few steps.
- For deployment purpose ,we created a service credentials in **Watson machine learning service**.
- After creating credentials load the model in service provider , We get the guid .

```
from watson_machine_learning_client import WatsonMachineLearningAPIClient

wml_credentials = {
    "apikey": "mMtFox3bgn8ZwUo80xuT_7uegZ06Hlwmdj55cde-x4Tg",
    "iam_apikey_description": "Auto-generated for key 22942c5d-09db-432c-aaa8-3769965759b9",
    "iam_apikey_name": "Service credentials-2",
    "iam_role_crn": "crn:v1:bluemix:public:iam::::serviceRole:Writer",
    "iam_serviceid_crn": "crn:v1:bluemix:public:iam-identity::a/1af47d91ccdc47489f9955bf340a1583::serviceid:ServiceId-a1a64fbd-7e6c-4f1b-9e25-799208ee5c1a",
    "instance_id": "fdc0cfe6-e3a5-4958-af00-d1453d727e98",
    "url": "https://eu-gb.ml.cloud.ibm.com"
}

client = WatsonMachineLearningAPIClient(wml_credentials)

metadata = {
    client.repository.ModelMetaNames.AUTHOR_NAME: 'Mohamed Abdullah',
    client.repository.ModelMetaNames.AUTHOR_EMAIL: 'abdullahmohamed2540@gmail.com',
    client.repository.ModelMetaNames.NAME: 'Prediction of Wind Direction',
    client.repository.ModelMetaNames.FRAMEWORK_NAME: 'tensorflow',
    client.repository.ModelMetaNames.FRAMEWORK_VERSION: '1.15',
    client.repository.ModelMetaNames.FRAMEWORK_LIBRARIES: [{'name': 'keras', 'version': '2.2.4'}]
}

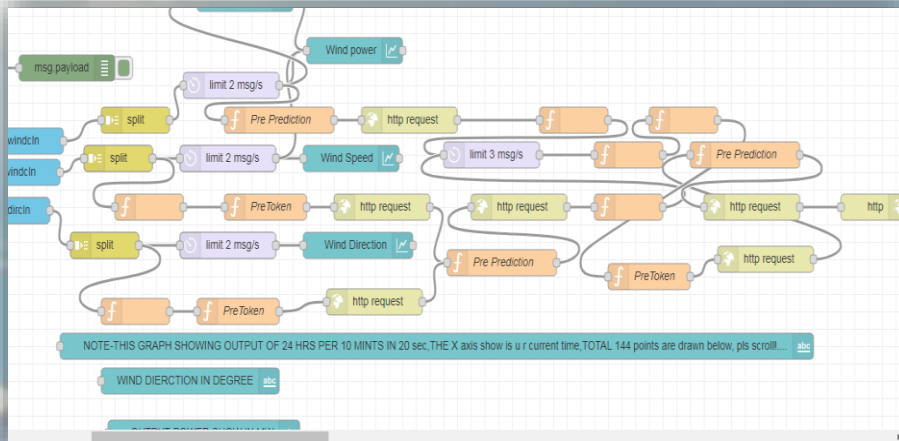
model_result_path = 'direction_prediction.h5'
lstm_model.save(model_result_path)

tar -zcvf direction_prediction.tgz direction_prediction.h5
direction_prediction.h5

stored_data = client.repository.store_model(model='direction_prediction.tgz', meta_props=metadata, training_data =X, training_target=y)
```

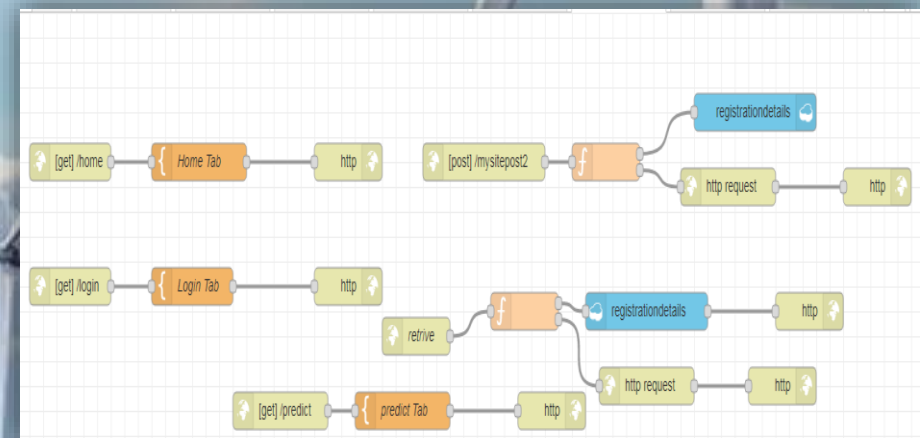

GUI Creation

For Creating GUI We used Node-red ,available in Our IBM .

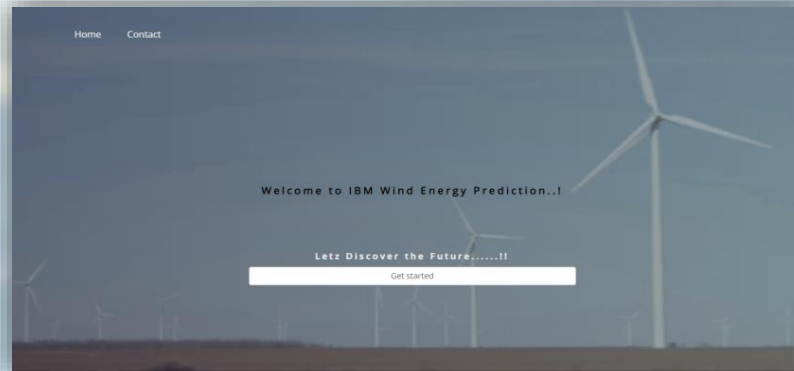


Prediction Flow

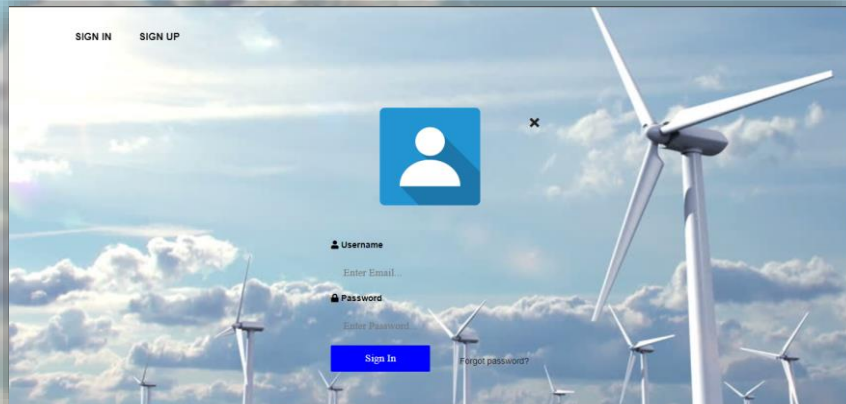
Main Flow



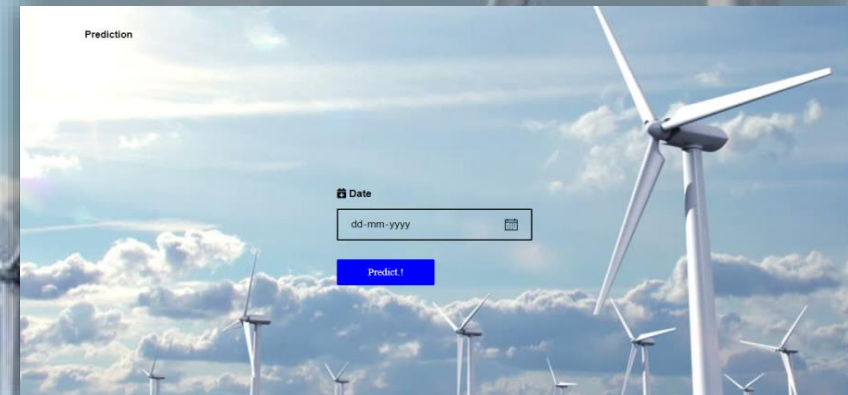
Our GUI With IBM



Home Tab



Log-In Tab



Prediction Tab

HIGHLIGHT'S

- We Predicting Wind Speed , Wind Direction In **separate algo.**, and giving the **output of this prediction** to predict **power**, however real time scenario happens like this.
- So our model should be best to **competitive with real time.**
- It can also **recommend** the time to **Utilize Power** from grid.
- So that we can give **rest period** to grid and **Save Energy** .
- **24 hours** forecast with **10 minutes** interval, just in **20 seconds**.

HIGHLIGHT'S

- For easy understanding of users “ Recommended Table Version ” are provided.
- To view forecast in better way, **Search** and **Filter** options are added.

RECOMMENDATION			
TABLE VERSION			
FILTER BY...			
WIND SPEED:	Filter By...	WIND DIRECTION:	Search By...
WIND POWER:	Filter By...		
TIME	WIND SPEED (M/S)	WIND DIRECTION (°)	WIND POWER (KW)
00:00:00	9.78811	67.52612	380.04779
00:10:00	10.09451	65.40338	3174.74194
00:20:00	10.14719	69.50209	3309.16894
00:30:00	9.97258	67.52629	3330.24609
00:40:00	10.29194	66.58721	3025.02295
00:50:00	10.71378	67.75638	2988.78906

RECOMMENDATION			
TIME	WIND SPEED (M/S)	WIND DIRECTION (°)	WIND POWER (KW)
00:00:00	9.78811	67.52612	380.04779
00:10:00	10.09451	65.40338	3174.74194
00:20:00	10.14719	69.50209	3309.16894
00:30:00	9.97258	67.52629	3330.24609
00:40:00	10.29194	66.58721	3025.02295
00:50:00	10.71378	67.75638	2988.78906
01:00:00	10.51755	65.80247	1261.66296
01:10:00	10.3856	66.7978	1307.474
01:20:00	10.22741	64.7504	1483.552

Future Plans

- To give **chart bots** to help more for the service users.
- Prediction of Seasons as well Location also help to maintain more stability in prediction
- All screen compatibility will be launched soon.



SPECIAL THANKS TO:-

- **IBM Platform And Crew**
- **SMART INTERNZ**

BY:-
MOHAMED HANEEF I
MOHAMED ABDULLAH K