IBM Recommendation Model To Utilize Energy From Wind Farm

-BY HOROSCOPER"S

SCOPE:-

- Our **Ulimate Aim** Is To **Competitive Real Time** , **Reduce Energy Wasteage** 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 Forecatsing Energy,so We Can **Save Machine** Break Down's.
- We Can Solve The **Future Energy Needs**.
- As Well **Availbiality Of Energy** Can Be Determined.

LITERATUER REVIEW:-

- We get into this great paper to get prerequisite in this domain.
- below mention's are our aspects from this paper

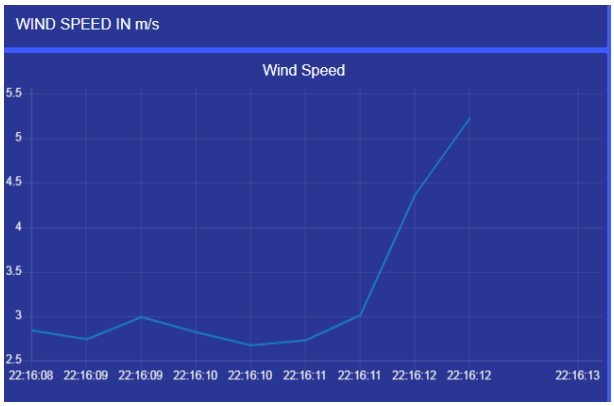
Article A New Hybrid Approach to Forecast Wind Power for Large Scale Wind Turbine Data Using Deep Learning with TensorFlow Framework and Principal Component Analysis

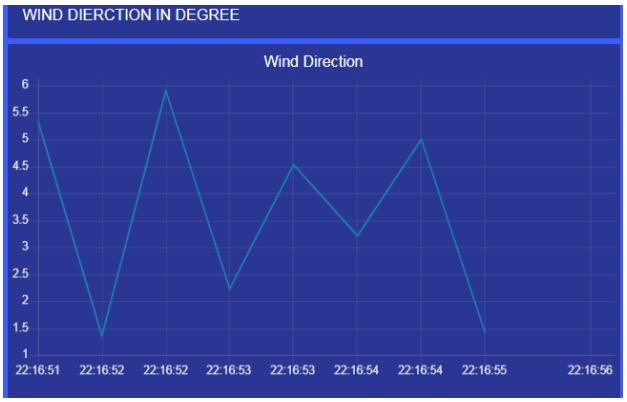
-Mansoor Khan , Tianqi Liu ,* and FarhanUllah

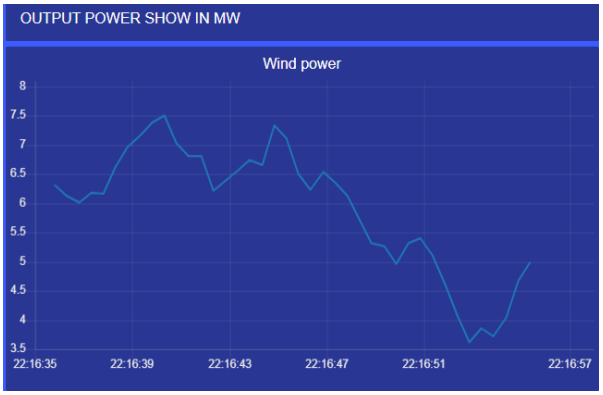
- The principal components are then used as input data to **deep learning** based on the TensorFlow framework. **PCA provides feature extraction and selection.**
- Here, **acc, val_loss, val_acc represents accuracy**, validation loss and validation accuracy, respectively. The loss, accuracy and loss, 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**.

REPORT ON -SBSPS_Challenge_2238-**IBM Recommendation Model To Utilize Energy From Wind Farm**

Forecasted Output Of Our Model:-







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

Work Process:-

- IBM Account Creation.
  - •Smart Internz assigning template.
  - •Assigning team.
- Fetching Input.
- Selection of algorithm.
- Date 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 **IBM watson studion** in a click.

```
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='HcWLNSvsl1J6y1Z0P9bQiEHBvJ5Ty-JY0OoShzpA1FiL',
    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-nxileyc1obvpgp',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,Autoarima** 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 Forescating** using **AutoAi**,The facility provied 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 **ouput** of **former prediction**.

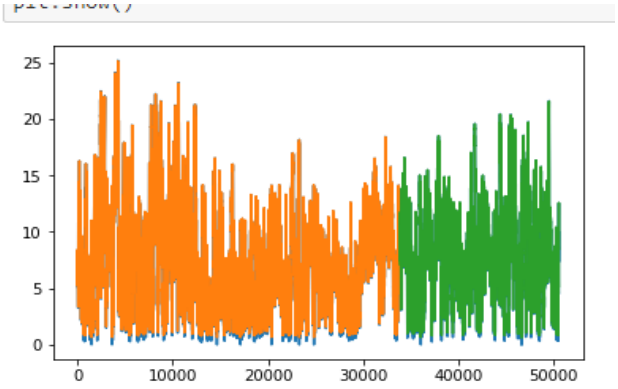
SPLITTING OF DATA

```
[8]: X = df[['WindSpeed','WindDirection']].values
     print(X)
     y = df['ActivePower'].values
     print(y)
     from sklearn.model_selection import train_test_split

     x_train,x_test,y_train,y_test = train_test_split(X,y, test_size = 0.2)
```

```
[[  5.31133604 259.99490356]
 [  5.67216682 268.64111328]
 [  5.2160368  272.56478882]
 ...
 [  8.43535805  84.74250031]
 [  9.42136574  84.2979126 ]
 [  9.97933197  82.27462006]]
[ 380.04779053  453.76919556  306.37658691 ... 2201.10693359 2515.6940918
 2820.46606445]
```

Validation of our Prediction:-

- The plot shows the **test and train dataed fitted with our model.**
- The RMSE the value showing below are **model fited by power** data which is especially done by **Auto Ai Service** By Our **IBM Watson Studio.**

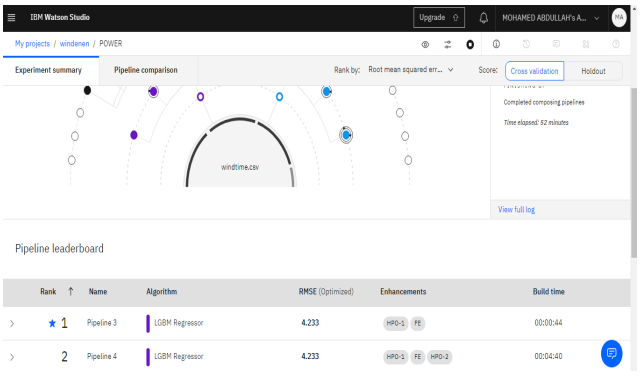


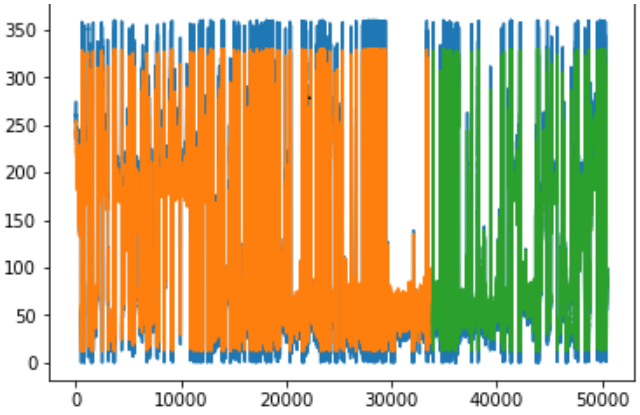

```
import math
# Estimate model performance
trainScore = model.evaluate(trainX, trainY, verbose=0
print('Train Score: %.2f MSE (%.2f RMSE)' % (trainSco
testScore = model.evaluate(testX, testY, verbose=0)
print('Test Score: %.2f MSE (%.2f RMSE)' % (testScore
```

```
Train Score: 0.58 MSE (0.76 RMSE)
Test Score: 0.54 MSE (0.74 RMSE)
```

```
model.predict([14.32318974])
```

```
array([[14.343465]], dtype=float32)
```





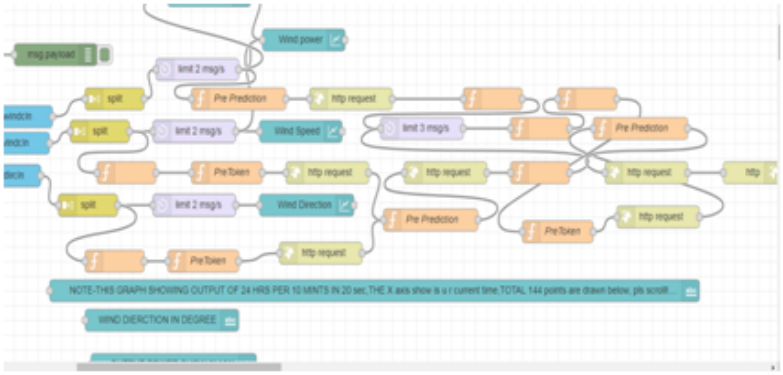
Deployment:-

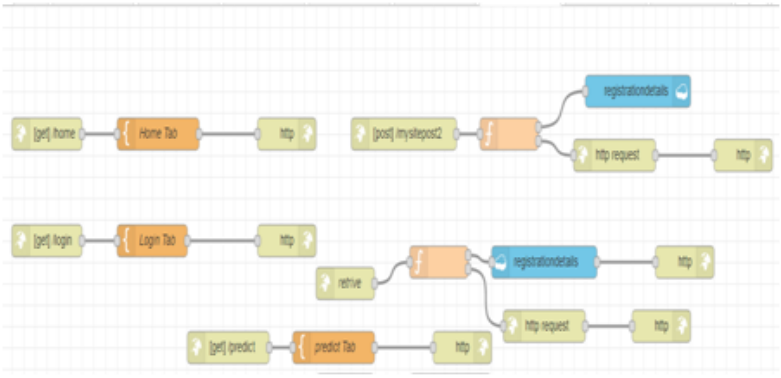
- **IBM Watson Studio** made **Deployment easy** in few steps.
- For deployment purpose ,we created a service credentials in **Watson mechaine learning service**.
- After creating credentials **load the model in service** provider,We **get the guid**
- that's all its done

```
from watson_machine_learning_client import WatsonMachineLearningAPIClient

wml_credentials = {
  "apikey": "mMtFox3bgn8Zx4oBOxuT_7uegZO6H8wmdjS5cde-x4Tg",
  "iam_apikey_description": "Auto-generated for key 22942c5d-09db-432c-aaa8-376096575969",
  "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-9e
25-799288ee5c1a",
  "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:-

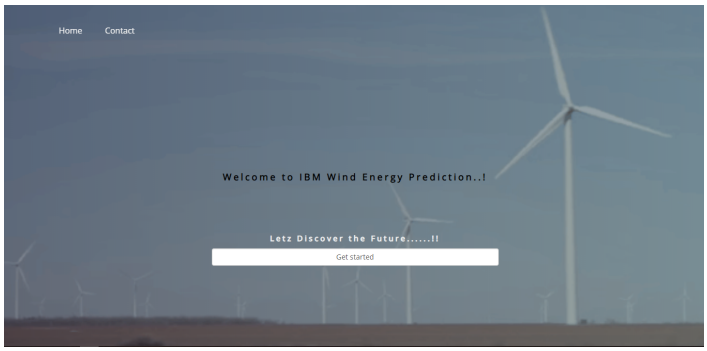
- We Done Our Gui With **Node-red**,it Is One Of **Service** Provied By **Our Ibm**
- Below Shown Node Are Our **Prediction Node** And Our **Gui Tab Nodes**

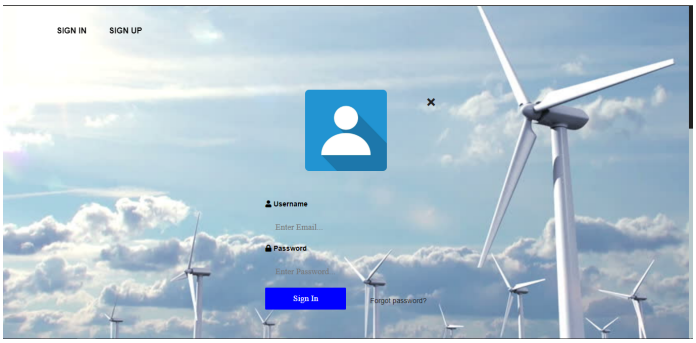


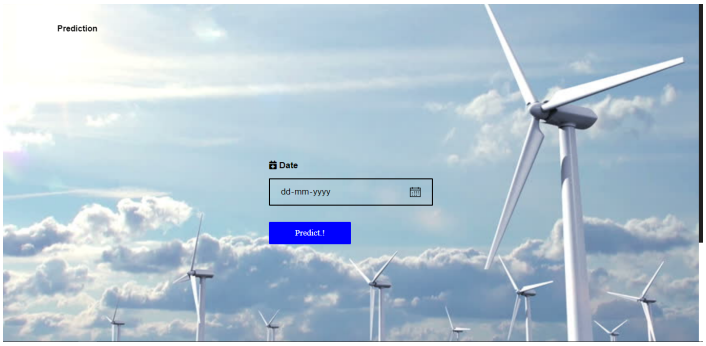


Our GUI With IBM

- ☞ below shown image's are our different tabs in our GUI.







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** .

Future Plans:-

- To give **Tabel verision** of our **IBM-Recommendation Model To Utilize Energy From Wind Farm**.For understand the prediction in better way.
- To give **chartbots** to **help more** for the service user's.
- **prediction** of **weather** also help to maintain **more stability** in prediction.

To Access our GUI:-

- https://node-red-itpxr.eu-gb.mybluemix.net/home

To Appreciate Our Work:-

- https://node-red-itpxr.eu-gb.mybluemix.net/home#contact

SPECIAL THANKS TO:-

IBM Platform And SMARTINTERNZ

by:-
Team -Horoscoper's
MOHAMED HANEEF I
MOHAMED ABDULLAH K