

Wind Turbine Power Prediction Problem Project Report

AI IN INDUSTRIES 4.0

Team :

- 1) Name - Anish Achintya
Roll - (SE20UARI020)
Email - anish20uari020@mahindrauniversity.edu.in**
- 2) Name - Mihir Ranjan
Roll - (SE20UARI095)
Email - mihir20uari095@mahindrauniversity.edu.in**
- 3) Osman Azam
Roll -(SE20UARI112)
Email - azam20uari112@mahindrauniversity.edu.in**

Please use the following link to see the resources output files and codes - <https://drive.google.com/drive/folders/1P3PMPAP9C5IkDpmq3iyoC6WPdJL2SYMV?usp=sharing>

Problem Statement

Forecast for Day Head Market -

- Forecast for next day (24hr ahead), how much power a wind turbine
- Calculate this forecast for the full year of data
- After the model is built, you can use only data from current point backwards in time.
- Future data should not be used During Testing and Running. All Data Can be use for building model, but testing and running only past data can be used
- Provide a new updated forecast every hour for a 24hour ahead forecast

Data Analysis and plotting

Our dataset consist of following columns:

- Date/Time - This contain date and time till 4 years. Data is collected at 10 min of time interval for 4 years.
- LV ActivePower (kW) - The power generated by the turbine for that moment .
- Wind Speed (m/s) - The wind speed at the hub height of the turbine (the wind speed that turbine use for electricity generation) .
- Theoretical_Power_curve (KWh) - The theoretical power values that the turbine generates with that wind speed which is given by the turbine manufacturer.
- Wind Direction (°) - The wind direction at the hub height of the turbine (wind turbines turn to this direction automatically)

Data Loading and Cleaning

Here we are dividing the dataset into three parts:

- Training - 60% of the dataset
- Validation - 20% of the dataset
- Testing - 20% of the dataset

Normalisation:

- We are using MiniMaxScaler to Normalise the features.

Feature Selection

Main Training model:

LSTM (Long Short Term Memory)

Note - Since Our corpus of data is very huge that is 210240 data points.

- Learning rate - 0.01 Learning rates that are too high may cause the model to converge too quickly, while rates that are too low may result in slow convergence or getting stuck in local minima.
- Optimiser - We are using ADAMS optimizer
- Batch Size - 32
- No of epoch - 50
- L1 Regularization

Model Selection

Below you can find different Model on which we can train our model. We are using LSTM (Long Short Term Memory) - We have a huge corpus of data i.e. 210240 data point

- Linear Regression:
 - Pros: Simple, interpretable, and fast.
 - Cons: Assumes a linear relationship between features and target.

- Random Forest Regression:
 - Pros: Handles non-linearity well, robust to outliers, and requires minimal feature scaling.
 - Cons: May be prone to overfitting, longer training times on large datasets.
- Gradient Boosting Regressor (e.g., XGBoost, LightGBM, CatBoost):
 - Pros: Excellent predictive performance, handles complex relationships, robust to outliers.
 - Cons: May require more tuning, longer training times.
- Neural Networks (Deep Learning):
 - Pros: Can capture complex patterns and relationships, suitable for large datasets.
 - Cons: May require more data, computational resources, and tuning.
- Support Vector Regression:
 - Pros: Effective in high-dimensional spaces, robust to outliers.
 - Cons: Sensitive to the choice of kernel and parameters.
- Autoregressive Integrated Moving Average (ARIMA):
 - Pros: Good for univariate time series forecasting, captures linear dependencies.
 - Cons: Assumes linear relationships, may not handle complex patterns well.
- Seasonal-Trend decomposition using LOESS (STL):
 - Pros: Decomposes time series into seasonal, trend, and residual components.
 - Cons: May require additional models for forecasting the individual components.
- Prophet:
 - Pros: Developed by Facebook, handles missing data and outliers well, captures seasonality and holidays.
 - Cons: May not perform well with datasets that have irregular holidays.

- Long Short-Term Memory (LSTM) Networks (Deep Learning):
 - Pros: Suitable for capturing long-term dependencies, handles non-linear patterns.
 - Cons: Requires more data, tuning, and computational resources.
- Seasonal Decomposition of Time Series (SARIMA):
 - Pros: Extension of ARIMA that includes seasonality components.
 - Cons: Assumes linear relationships.
- Exponential Smoothing State Space Models (ETS):
 - Pros: Good for univariate time series forecasting, includes error, trend, and seasonality components.
 - Cons: Limited to univariate time series.

Once we selected out model -

- LSTM Model Building: Creating a Sequential model with an LSTM layer, a Dense layer, and regularization.
- Model Compilation and Training: Compiling the model with Adam optimizer, mean squared error loss, and mean absolute error metric.
- Training the model on the training set and validating on the validation set.
- Model Evaluation on Test Set: Evaluating the model on the test set and printing the mean absolute error.

Testing

Below is the result we got from testing the test dataset 20% of our main dataset i.e. dataset.csv

Test Mean Absolute Error: 152.20657348632812

45/45 [=====] - 0s 1ms/step

	Actual	Predicted	RMSE
35410	720.594482	2.178398e+06	6.668939e+06
23250	632.309814	2.379462e+06	6.668939e+06
35570	470.704193	1.712162e+06	6.668939e+06
11715	1920.000000	6.645934e+06	6.668939e+06
40256	556.119385	2.066000e+06	6.668939e+06
...
2254	3414.837891	1.243454e+07	6.668939e+06
6224	1328.156982	3.682776e+06	6.668939e+06
33427	215.810501	8.227719e+05	6.668939e+06
36504	3389.184082	1.181621e+07	6.668939e+06
28445	0.000000	2.881901e+03	6.668939e+06