WIND SPEED PREDICTION

A
Project Report
Submitted in partial fulfilment of the
Requirements for the award of the Degree
of

BACHELOR OF ENGINEERING
IN
INFORMATION TECHNOLOGY

By

NARSINI SHRAVANI 1602-20-737-039

THODETI SHIVA CHARAN 1602-20-737-038

ZOHA TABASSUM 1602-20-737-053

Under the guidance of

Mrs. Swapna Sri

Associate Professor

Department of Information Technology

Vasavi College of Engineering (Autonomous)

Accredited by NAAC with 'A++' Grade

(Affiliated to Osmania University) Ibrahimbagh,

Hyderabad-500 031

Vasavi College of Engineering (Autonomous)
Accredited by NAAC with 'A++' Grade (Affiliated to Osmania University) Hyderabad-500 031
Department of Information Technology

DECLARATION BY THE CANDIDATES

We, **Narsini Shravani, Shiva Charan& Zoha Tabassum,** bearing hall ticket numbers, **1602-20-737-039, 1602-20-737-038 & 1602-20-737-53** respectively, hereby declare that the project report entitled **Windspeed Prediction** under the guidance of **Mrs.Swapna Sri,** Associate Professor, Department of Information Technology, Vasavi College of Engineering, Hyderabad, is submitted in partial fulfilment of the requirement for the award of the degree of **Bachelor of Engineering** in **Information Technology**

This is a record of bonafide work carried out by me and the results embodied in this project report have not been submitted to any other university or institute for the award of any other degree or diploma.

Narsini Shravani
1602-20-737-039.

Thodeti Shiva Charan
1602-20-737-038.

Zoha Tabassum
1602-20-737-053.

Vasavi College of Engineering (Autonomous)

Accredited by NAAC with 'A++' Grade

(Affiliated to Osmania University)

Hyderabad-500 031

Department of Information Technology

DECLARATION BY THE CANDIDATE

I**, Narsini Shravani,** bearing hall ticket number, **1602-20-737-039**, hereby declare that the project report entitled **Windspeed Prediction** under the guidance of **Mrs.Swapna Sri,**Associate Professor, Department of Information Technology, Vasavi College of Engineering, Hyderabad, is submitted in partial fulfilment of the requirement for the award of the degree of **Bachelor of Engineering** in **Information Technology**.

This is a record of bonafide work carried out by me and the results embodied in this project report have not been submitted to any other university or institute for the award of any other degree or diploma.

Narsini Shravani.
1602-20-737-039

Vasavi College of Engineering (Autonomous)

Accredited by NAAC with 'A++' Grade

(Affiliated to Osmania University)

Hyderabad-500 031

Department of Information Technology

DECLARATION BY THE CANDIDATE

I, **Shiva Charan,** bearing hall ticket number, **1602-20-737-038**, hereby declare that the project report entitled **Windspeed Prediction** under the guidance of **Mrs.Swapna Sri,** Associate Professor, Department of Information Technology, Vasavi College of Engineering, Hyderabad, is submitted in partial fulfilment of the requirement for the award of the degree of **Bachelor of Engineering** in **Information Technology**.

This is a record of bonafide work carried out by me and the results embodied in this project report have not been submitted to any other university or institute for the award of any other degree or diploma.

Shiva Charan.
1602-20-737-038

Vasavi College of Engineering (Autonomous)

Accredited by NAAC with 'A++' Grade

(Affiliated to Osmania University)

Hyderabad-500 031

Department of Information Technology

DECLARATION BY THE CANDIDATE

I, **Zoha Tabassum,** bearing hall ticket number, **1602-20-737-053**, hereby declare that the project report entitled **Windspeed Prediction** under the guidance of **Mrs.Swapna Sri,** Associate Professor, Department of Information Technology, Vasavi College of Engineering, Hyderabad, is submitted in partial fulfilment of the requirement for the award of the degree of **Bachelor of Engineering** in **Information Technology**.

This is a record of bonafide work carried out by me and the results embodied in this project report have not been submitted to any other university or institute for the award of any other degree or diploma.

Zoha Tabassum.
1602-20-737-053

Vasavi College of Engineering (Autonomous)

Accredited by NAAC with 'A++' Grade

(Affiliated to Osmania University) Hyderabad-500 031

Department of Information Technology

BONAFIDE CERTIFICATE

This is to certify that the project entitled **Windspeed Prediction** being submitted by Narsini Shravani, Shiva Charan& Zoha Tabassum bearing **1602-20-737-039, 1602-20-737-038 & 1602-20-737-053** in partial fulfilment of the requirements for the award of the degree of Bachelor of Engineering in Information Technology is a record of bonafide work carried out by him/her undermy guidance.

Mrs.Swapna Sri	**External Examiner**	**Dr. K. Ram Mohan Rao**
Associate Professor		**HOD, IT**
Internal Guide		

ACKNOWLEDGEMENT

The satisfaction that accompanies that the successful completion of the project would not have been possible without the kind support and help of many individuals. We would like to extend my sincere thanks to all of them. We would like to take the opportunity to express our humble gratitude to **Mrs.Swapna Sri**, Associate Professor under whom we executed this project. We are grateful to her guidance, and constructive suggestions that helped us in the preparation of this project. Her constant guidance and willingness to share her vast knowledge made us understand this project and its manifestations in great depths and helped us to complete the assigned tasks. We would like to thank all faculty members andstaff of the Department of Information Technology for their generous help in various ways for the completion of this project. Finally, yet importantly, We would like to express our heartfelt thanks to our Head of the Department **Dr. K. Ram Mohan Rao** Sir

TABLE OF CONTENTS:

ABSTRACT

Wind speed prediction using CNN-LSTM is a deep learning technique that uses a combination of convolutional neural networks (CNN) and long short-term memory (LSTM) models to predict future wind speed values based on historical wind speed data. This technique is important for various applications such as renewable energy production and weather forecasting.

Renewable energy sources, nuclear energy and fossil fuels, are the three main types of energy used to generate electricity. The majority of electricity is generated by steam turbines powered by nuclear, fossil fuels, geothermal, biomass, and solar thermal energy. As wind turbines, gas turbines, hydro turbines and solar PV are of the major energy technologies. As fossil fuels and nuclear energy are non-renewable energy resources, we need to switch for renewable energy resources where wind energy plays a vital role.

Wind energy is fastest growing renewable energy source and is one of the emerging sustainable sources of electricity. Wind energy production and grid integration are becoming extremely important. Accurate wind speed predictions can help improve the efficiency of wind turbines, optimize energy production, and prevent potential disasters caused by extreme wind conditions.

CHAPTER 1

INTRODUCTION

What is wind speed prediction?

Wind speed prediction is the process of predicting future wind speed based on the historic wind speed data.

1.1 PURPOSE

Wind energy is one of the important renewable energy resource. With wind we can generate the electricity. The purpose of our project is to predict the most accurate values so that it helps the energy producers to plan their energy generation and distribution accordingly .

1.2 PRODUCT SCOPE

Wind energy plays an important role in generating electricity and has various usecases. So it is important to predict them correctly. Our model plays an important role in predicting it.

1.3 PROBLEM DEFINATION

Wind speed prediction using CNN-LSTM is a deep learning technique that uses a combination of convolutional neural networks (CNN) and long short-term memory (LSTM) models to predict future wind speed values based on historical wind speed data. This technique is important for various applications such as renewable energy production and weather forecasting. Accurate wind speed predictions can help improve the efficiency of wind turbines, optimize energy production, and prevent potential disasters caused by extreme wind conditions.

CHAPTER 2

RELATED WORK

Wind speed prediction using CNN-LSTM models offers several advantages over traditional methods. Here are some of the key advantages:

1. Ability to capture spatial and temporal patterns: CNN-LSTM models can capture both spatial and temporal dependencies in wind speed data. Convolutional layers in the CNN component can effectively extract spatial features, such as local patterns and spatial correlations, from the input data. LSTM layers, on the other hand, can capture temporal dependencies and long-term patterns. By combining both components, CNN-LSTM models can better model the complex relationships between wind speed and its spatial and temporal characteristics.

2. Automatic feature extraction: CNN-LSTM models have the advantage of automatically learning relevant features from the input data. Traditional methods often require manual feature engineering, where domain experts need to handcraft relevant features from the raw data. CNN-LSTM models, on the other hand, can automatically learn and extract features from the input data, reducing the need for manual feature engineering and potentially capturing more intricate patterns that might be overlooked by human-designed features.

3. Nonlinear modeling capability: Wind speed prediction can be influenced by various complex nonlinear relationships. CNN-LSTM models, with their deep learning architecture, have the ability to model and capture these nonlinear relationships effectively. Traditional methods, such as linear regression or autoregressive models, may struggle to capture the complex nonlinear dynamics present in wind speed data.

4. Adaptability to varying input formats: CNN-LSTM models can handle different input formats, such as multi-dimensional grids or sequential time series. This flexibility allows them to accommodate diverse types of wind speed data, such as data from meteorological stations, remote sensing instruments, or weather prediction models. Traditional methods often have specific assumptions about data format or require data preprocessing to fit into a particular format, limiting their applicability to different data types.

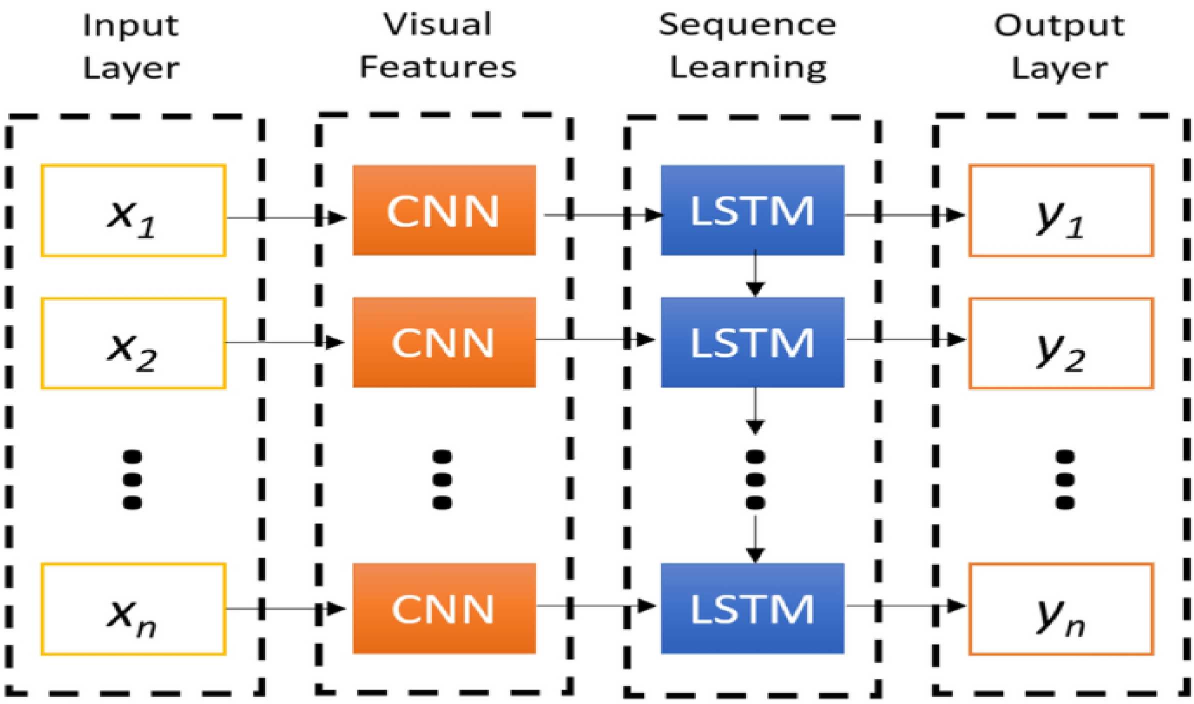
5. Improved prediction accuracy: CNN-LSTM models have shown promising results in wind speed prediction, often outperforming traditional methods in terms of prediction accuracy. The ability to capture spatial and temporal patterns, along with automatic feature extraction and nonlinear modeling capabilities, can lead to more accurate forecasts. Improved accuracy in wind speed prediction can have practical implications in various domains, such as renewable energy management, aviation, and environmental planning.

While CNN-LSTM models offer these advantages, it's important to note that the performance of any predictive model depends on several factors, including the quality and quantity of the available data, model architecture, hyperparameter tuning, and training methodology. Careful consideration and experimentation are necessary to achieve optimal results when applying CNN-LSTM models for wind speed prediction.

CHAPTER 3

PROPOSED WORK

3.1ARCHITECTURE



The CNN-LSTM model is built of two layers: the CNN layer and the LSTM layer

3.2 Technologies used –

Technology used –The tool using which the ‘Wind speed prediction’ was made is google colab.

3.3 Implementation:

Firstly we feed the data into the CNN layers. Where we create the class by using the function called sequential () and we define the number of layers through which our data is passed.

In our approach we have defined five layers in which the First Layer: con1D, Second Layer: MaxPooling1D, Third Layer: Flatten, Fourth Layer: Dense (activation = ‘ReLu’), Fifth Layer: Dense.

The input data is given to cnn. The main function of a 1D convolutional neural network (Conv1D) is to extract useful features from one-dimensional input data, such as time-series data

The output of a 1D convolutional layer can be fed into other layers, such as pooling layers or dense layers

MaxPooling1D is used after a Conv1D layer, to reduce the dimensionality of the output and to extract the most important features from the convolutional layer.

Flatten() is a Keras layer that flattens the input into a 1D array. It is typically used after convolutional layers and pooling layers, to convert the output of these layers into a 1D vector that can be fed into a fully connected (dense) layer.

The Dense layer is also known as a fully connected layer,A Dense layer is a layer where all the neurons in the previous layer are connected to all the neurons in the current layer. Each neuron in the Dense layer takes as input a vector of features that has been extracted by the previous layers, and produces a single output value. The main purpose of the Dense layer in a CNN is to perform classification or regression on the features extracted by the convolutional and pooling layers.

Now we get the featured output and we pass it into the lstm. The output of CNN layer is given to the LSTM where training of model is done, and prediction is made. Going deep into the LSTM part where we added the layers to a class by using sequential () function in that we added fours layers, they are three lstm layers and one dense layer for output purpose.

The purpose of LSTM layers is to model sequences of data that have some temporal relationship between the data points. LSTMs can capture long-term dependencies in the input data and are widely used in applications such as speech recognition, nlp, and time-series forecasting.

3.3.1 MODULES:

Numpy:

The NumPy module is a fundamental package in Python for scientific computing and working with arrays. It provides powerful mathematical functions and operations on multi-dimensional arrays, along with a collection of tools for working with these arrays efficiently

Pandas:

The pandas module is a powerful library in Python for data manipulation and analysis. It provides easy-to-use data structures, such as DataFrame and Series, along with a wide range of functions and methods for efficiently working with structured data.

Matplotlib:

Matplotlib is a comprehensive library for creating static, animated, and interactive visualizations in Python. Matplotlib makes easy things easy and hard things possible.

Create publication quality plots.

Make interactive figures that can zoom, pan, update.

Customize visual style and layout.

Export to many file formats.

Embed in Jupyter Lab and Graphical User Interfaces.

TensorFlow:

TensorFlow is an open-source library for fast numerical computing. It was created and is maintained by Google and released under the Apache 2.0 open-source license. The API is nominally for the Python programming language, although there is access to the underlying C++ API. Unlike other numerical libraries intended for use in Deep Learning like Theano, TensorFlow was designed for use both in research and development and in production systems, not least Rank Brain in Google search and the fun project. It can run on single CPU systems, GPUs as well as mobile devices and large-scale distributed system of hundreds of machines.

3.3.2 ALGORITHMS USED:

CNN:

The Convolutional Neural Network (CNN) algorithm is a deep learning algorithm specifically designed for analyzing visual data, such as images. CNNs have revolutionized the field of computer vision and achieved remarkable success in various tasks, including image classification, object detection, and image segmentation.

LSTM:

The Long Short-Term Memory (LSTM) algorithm is a type of recurrent neural network (RNN) architecture that is particularly effective in handling sequence data, such as time series, text, and speech. LSTMs were designed to address the vanishing gradient problem faced by traditional RNNs, allowing them to capture long-term dependencies in sequences.

3.3.3 CODE AND GITHUB LINKS

CODE:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

from google.colab import drive
drive.mount('/content/drive')

path="/content/drive/MyDrive/project/winddata2010.csv"
dataset=pd.read_csv(path,engine='python' )



data=dataset.reset_index()['wind speed at 100m (m/s)']
```

```
data=data[0:4320]

data=data.reshape(-1,1)

from sklearn.preprocessing import MinMaxScaler
scaler=MinMaxScaler(feature_range=(0,1))
data=scaler.fit_transform(data)

train_size=int(len(data))*0.80
train_size=int(train_size)
test_size=len(data)-train_size
test_size=int(test_size)

traindata=data[0:train_size,:]
testdata=data[train_size:len(data),:1]


def create(dataset,timesteps):
    datax,datay=[],[]
    for i in range(len(dataset)-timesteps-1):
        a=dataset[i:(i+timesteps),0]
        datax.append(a)
        datay.append(dataset[i+timesteps,0])
    return np.array(datax),np.array(datay)


time_step=288
xtrain,ytrain=create(traindata,time_step)
xtest,ytest=create(testdata,time_step)


xtrain.shape
```

```
test_size
xtrain.shape[0]
xtrain=xtrain.reshape(xtrain.shape[0],288,1)
xtest=xtest.reshape(xtest.shape[0],288,1)
import warnings
import time
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from keras import optimizers
from keras.models import Sequential, Model
from keras.layers.convolutional import Conv1D, MaxPooling1D
from keras.layers import Dense, LSTM, RepeatVector, TimeDistributed, Flatten

model_cnn = Sequential()
model_cnn.add(Conv1D(filters=128, kernel_size=1, activation='relu',
input_shape=(xtrain.shape[1],1)))
model_cnn.add(MaxPooling1D(pool_size=1))
model_cnn.add(Flatten())
model_cnn.add(Dense(50, activation='relu'))
model_cnn.add(Dense(1))
model_cnn.compile(loss='mse', optimizer='adam')
model_cnn.summary()

cnn_history = model_cnn.fit(xtrain, ytrain,validation_data=(xtest,ytest),epochs=10,batch_size=32)


plt.plot(cnn_history.history['val_loss'],label='loss')
plt.plot(cnn_history.history['loss'],label='loss')
plt.title('Epochs')
plt.show()
```

```
y2=model_cnn.predict(xtest)
y2.shape


y2=scaler.inverse_transform(y2)
print(y2[1:10])
ytest=ytest.reshape(-1,1)
ytest=scaler.inverse_transform(ytest)


plt.figure(dpi=150)
plt.plot(y2,label='pred')
plt.plot(ytest,label='actual')
plt.legend()


def accuracy(pre,act):
    mse=abs(pre-act).mean()
    print(mse)


accuracy(y2,ytest)

features = Model(inputs=model_cnn.inputs, outputs=model_cnn.layers[3].output)
train= features.predict(xtrain)
test = features.predict(xtest)


from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense
from tensorflow.keras.layers import LSTM
```

```
from tensorflow.keras.layers import Dropout


model.add(LSTM(50,return_sequences=True,input_shape=(xtrain.shape[1],1)))

model.add(LSTM(100,return_sequences=True))

model.add(LSTM(50))

model.add(Dense(1))
model.summary()


model.compile(loss='mse',optimizer='adam')

plt.plot(history.history['val_loss'],label='loss')
plt.plot(history.history['loss'],label='loss')
plt.title('Epochs')
plt.show()


y2=model.predict(xtest)

y2=y2.reshape(-1,1)
y2=scaler.inverse_transform(y2)




plt.figure(dpi=150)
plt.plot(y2,label='pred')
plt.plot(ytest,label='actual')
plt.legend()
```

```
plt.xlabel('time instance')
plt.ylabel('wind speed in m/s')


def accuracy(pre,act):
    mse=abs(pre-act).mean()
    print(mse)

accuracy(y2,ytest)
```

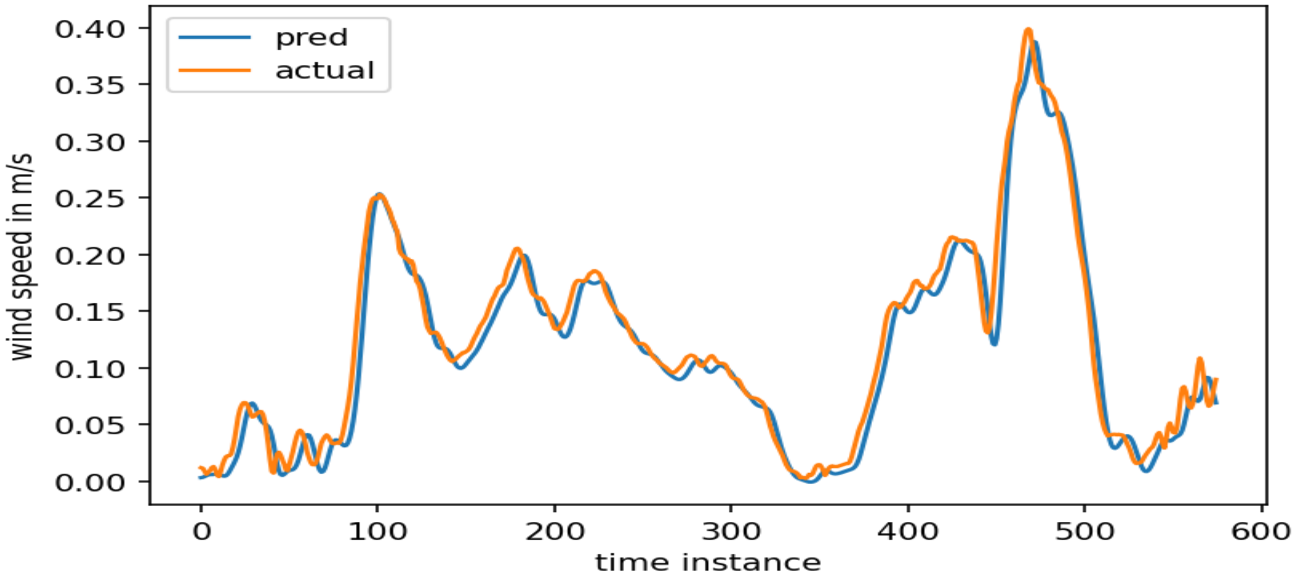
3.5.3 – GitHub Links –

https://github.com/ShravaniNarsini/WindSpeed-Prediction

CHAPTER 4 –

RESULTS

The following are the results obtained after implementation –



CHAPTER 5

DISCUSSION AND FUTURE WORK –

Prediction of wind speed is used to optimize wind energy and it is crucial for planning of wind power and for the stable and uninterrupted operation of the power system. This paper proposes CNN-LSTM based hybrid wind speed forecasting approach. The proposed framework receives real time data from NREL for the prediction of wind speed. Experimental results reveal that the proposed model has best approximation as compared to other Deep Learning models in terms of RMSE, MSE, and MAE. CNN is utilized to extract features from the data, while LSTM is adopted to predict the wind speed with extracted feature data. According to the results of the experiment, the proposed framework performed well than all other approaches in terms of prediction performance. The proposed model had the least number of errors.

CHAPTER 6

REFERENCES

[1] Kanna Bhaskar, S.N.Singh, "AWNN-Assisted Wind Power Forecasting Using Feed-Forward Neural Network," in *IEEE Transactions on Sustainable Energy*, vol.3,No.2,April 2012,doi:10.1109/TSTE.2011.2182215

[2] Mahdi Khodayar, Okyay Kaynak and Mohammad E. Khodayar, "Rough Deep Neural Architecture for Short-Term Wind Speed Forecasting," *IEEE Transactions on Industrial Informatics, Vol.13, No.6, December 2017,* doi: 10.1109/TII.2017.2730846

[3] Madasthu Santhosh, Chintham Venkaiah, D.M. Vinod Kumar, "Ensemble empirical mode decomposition based adaptive wavelet neural network method for wind speed prediction," in *Energy Conversion and Management 168(2018) 482-493,* doi: 10.1016/j.enconman.2018.04.099

[4] Vikram Bali, Ajay Kumar, Satyam Gangwar, "Deep Learning based Wind Speed Forecasting- A Review," in 9th *International Conference on Cloud Computing, Data Science & Engineering,2019.*