Weather Forecasting using Deep Learning

ABSTRACT

The main objective of the project is to design and development of A system for weather forecasting. Traditionally, weather predictions are performed with the help of large complex models of physics, which utilize different atmospheric conditions over a long period of time. We present a weather prediction technique that utilizes historical data of Delhi's weather and predicting using Convolutional LSTM Neural Network.

INTRODUCTION

Weather conditions around the world change rapidly and continuously. Correct forecasts are essential in today's daily life. From agriculture to industry, from traveling to daily commuting, we are dependent on weather forecasts heavily. As the entire world is suffering from the continuous climate change and its side effects, it is very important to predict the weather without any error to ensure easy and seamless mobility, as well as safe day to day operations. Weather forecasting is the application of science and technology to predict the conditions of the atmosphere for a given location and time. Weather forecasts are made by collecting quantitative data about the current state of the atmosphere at a given place and using meteorology to project how the atmosphere will change. In this 21st century, weather forecasting holds great importance and is used in several areas ranging from keeping track of agricultural field weather conditions to that of industrial conditions monitoring. Weather Forecasting would help in keeping record of different climatic behaviors which includes temperature, wind, humidity, etc.

Here in this project we are trying to predict the weather of New Delhi, India. The dataset used in this project contains the different features related to weather form year 1996 to year 2017. It contain many different features like date, time, wind, temperature, humidity, climate, weather, etc. The main technique used in is project is Time Series Forecasting.

LITERATURE REVIEW

The weather prediction application involves two major components: the sliding window algorithm and ID3 algorithm. The sliding window algorithm divides a collection of data into groups. More specifically, the sliding window algorithm is used to segment a two-week collection of historical weather data into week long windows. Machine learning in weather forecasting is a recent trend in the literature. There are several works which discuss this topic. Holmstrom. Proposed a technique to forecast the maximum and minimum temperature of the next seven days, given data of past two days [6]. They utilized a linear regression model, as well as a variation of a functional linear regression model. They showed that both the models were out performed by professional weather forecasting services for the prediction of up to seven days. However, their model performs better in forecasting later days or longer time scales. A hybrid model that used neural networks to model the physics behind weather forecasting was proposed by Krasnopolsky and Rabinivitz[7]. Support vector machine was utilized for weather prediction as a classification problem by Radhika et al.[9]. A data mining based predictive model to identify the fluctuating patterns of weather conditions was proposed[11].The patterns from historical data is used to approximate the upcoming weather conditions. The proposed data model uses Hidden Markov Model for prediction and k-means clustering for extracting weather condition observations. Grover et al. studied weather prediction via a hybrid approach, which combines discriminatively trained predictive models with deep neural networks that models the joint statistics of a set of weather-related variables[5]. Montori et al. used the concept of crowdsensing, where participating users share their smart phone data to environmental phenomenons [8]. They introduced an architecture named SenSquare, which handles data from IoT sources and crowdsensing platforms, and display the data unifiedly to subscribers. This data is used in smartcity environment monitoring. However, none of these work use the idea of combining data from neighboring places.

DATASET

The dataset which is used contains weather data for New Delhi, India.This data was taken out from wunderground with the help of their easy to use API.It contains various features such as temperature, pressure, humidity, rain, precipitation, etc.This data is owned by wunderground and although I ended up using noaa's data for my research, I thought that I'd share this data here as I haven't worked on hourly data yet and this might be of huge importance.The main target is to develop a prediction model accurate enough for predicting the weather. We can try something like predicting the weather in the next 24 hours like Microsoft tried some time back.

METHODOLOGY

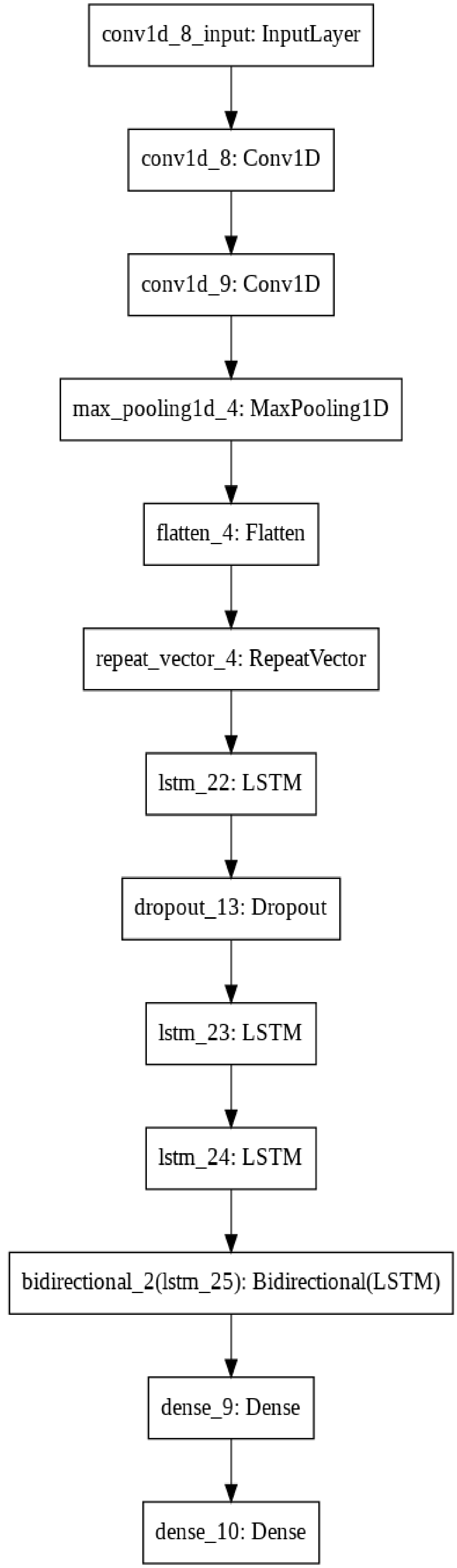
First, we are analyzing the weather conditions in Delhi for example, which type of whether is most common in Delhi, the most common temperature range in the city, temperature and humidity condition throughout the year etc. Then we are performing Time Series Analysis on the time series data of temperature in Delhi.

Time series analysis comprises methods for analyzing time series data in order to extract meaningful statistics and other characteristics of the data. Time series forecasting is the use of a model to predict future values based on previously observed values. Time series are widely used for non-stationary data, like economic, weather, stock price, and retail sales in this post. We are going to predict temperature with time series forecasting using RNN method.

Our forecasting model comprises multiple Convolutional and LSTM layers. Long Short Term Memory networks – usually just called "LSTMs" – are a special kind of RNN, capable of learning long-term dependencies. They were introduced by Hochreiter & Schmidhuber (1997), and were refined and popularized by many people in following work. They work tremendously well on a large variety of problems, and are now widely used. LSTMs are explicitly designed to avoid the long-term dependency problem. Remembering information for long periods of time is practically their default behavior. All recurrent neural networks have the form of a chain of repeating modules of neural network. In standard RNNs, this repeating module will have a very simple structure, such as a single *tanh* layer.

Here is the summary and visualization of our model:

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv1d (Conv1D)              (None, 29, 256)           768
_________________________________________________________________
conv1d_1 (Conv1D)            (None, 28, 128)           65664
_________________________________________________________________
max_pooling1d (MaxPooling1D) (None, 14, 128)           0
_________________________________________________________________
flatten (Flatten)            (None, 1792)              0
_________________________________________________________________
repeat_vector (RepeatVector) (None, 30, 1792)          0
_________________________________________________________________
lstm (LSTM)                  (None, 30, 100)           757200
_________________________________________________________________
dropout (Dropout)            (None, 30, 100)           0
_________________________________________________________________
lstm_1 (LSTM)                (None, 30, 100)           80400
_________________________________________________________________
lstm_2 (LSTM)                (None, 30, 100)           80400
_________________________________________________________________
bidirectional (Bidirectional (None, 256)               234496
_________________________________________________________________
dense (Dense)                (None, 100)               25700
_________________________________________________________________
dense_1 (Dense)              (None, 1)                 101
=================================================================
Total params: 1,244,729
Trainable params: 1,244,729
Non-trainable params: 0
_________________________________________________________________
```



First we are taking two convolution layers with kernel size 2 by 2 in both layers. The we added a Max Pooling Layer of pool size 2 by 2 to drop some values and take more relevant of them for further process. Then we are Flattening the data and repeating the vectors 30 times to feed in further LSTM layers. Then we are adding a LSTM layer with activation function tanh then adding a dropout layer to drop 20% values and then data goes to further two identical LSTM layers with activation function tanh. Then adding a Bidirectional LSTM layer with activation function relu.

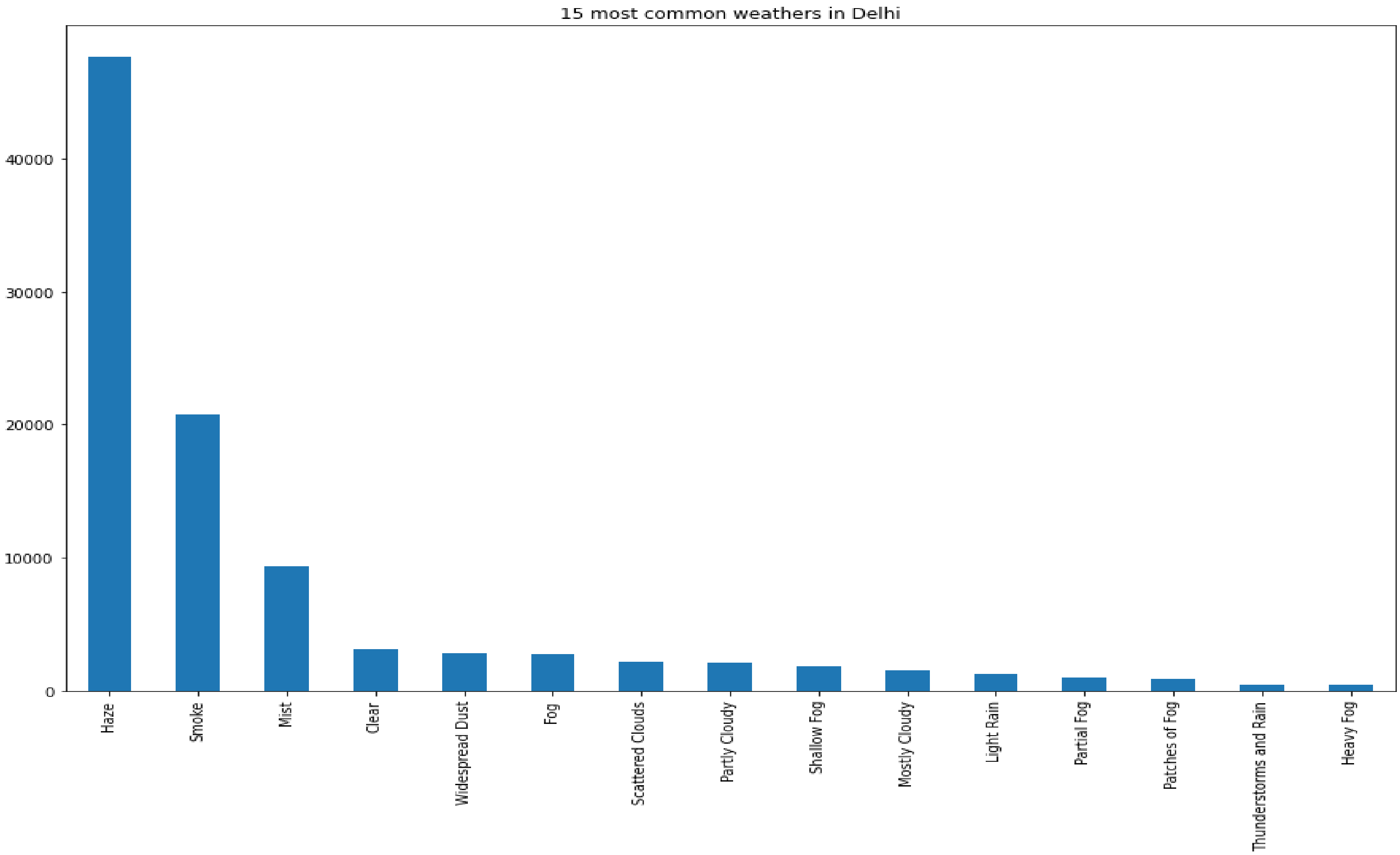
Using bidirectional will run your inputs in two ways, one from past to future and one from future to past so using the two hidden states combined you are able in any point in time to preserve information from both past and future. This can be beneficial for out forecasting model.

Then we added a normal ANN layer with activation function with relu activation function and then added an output node to give the output.

We compiled the model with ADAM optimizer. ADAM is a modified version of Stochastic Gradient Descent optimizer and it is proved in many researches that this works really well with Deep Learning models. We trained out model with 300 epochs.

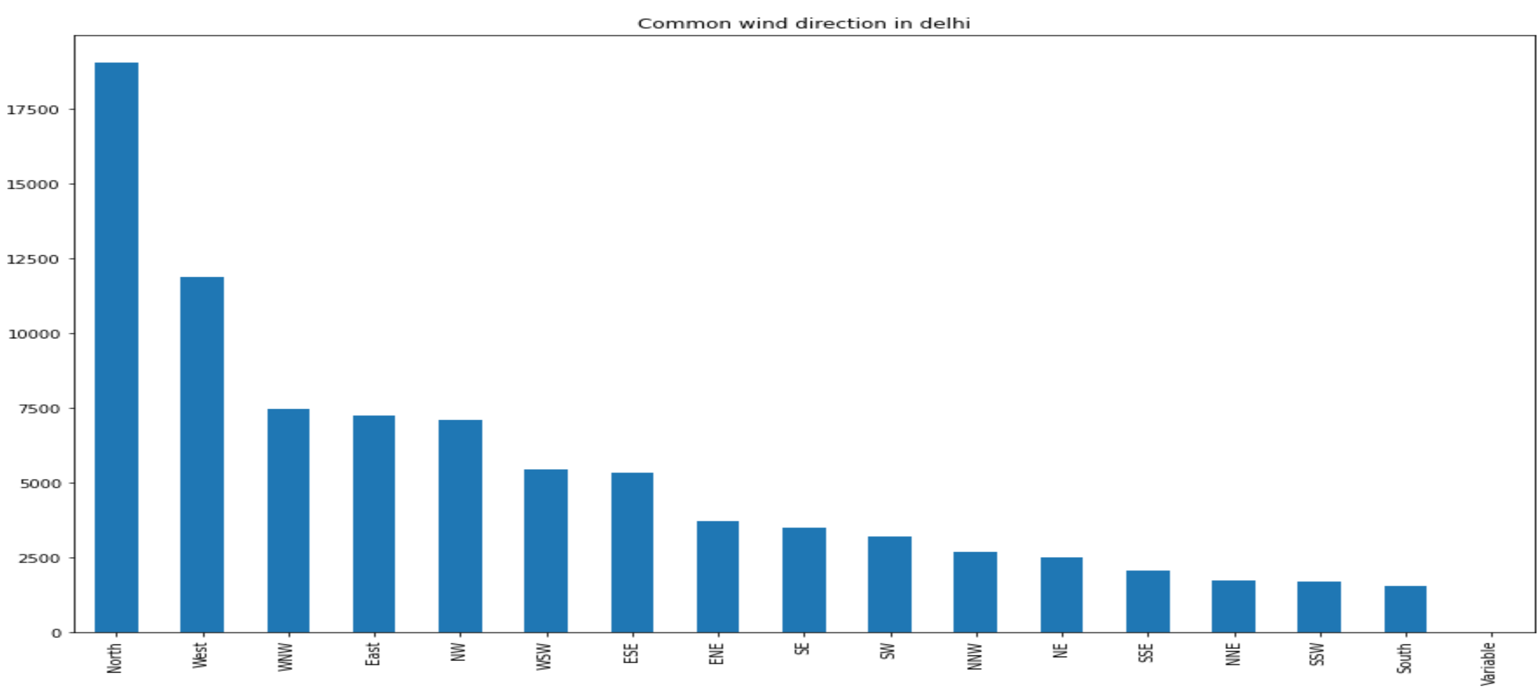
RESULT AND DISCUSSION

First we analyzed the data and checked what are the most common weather conditions in Delhi. The result is :



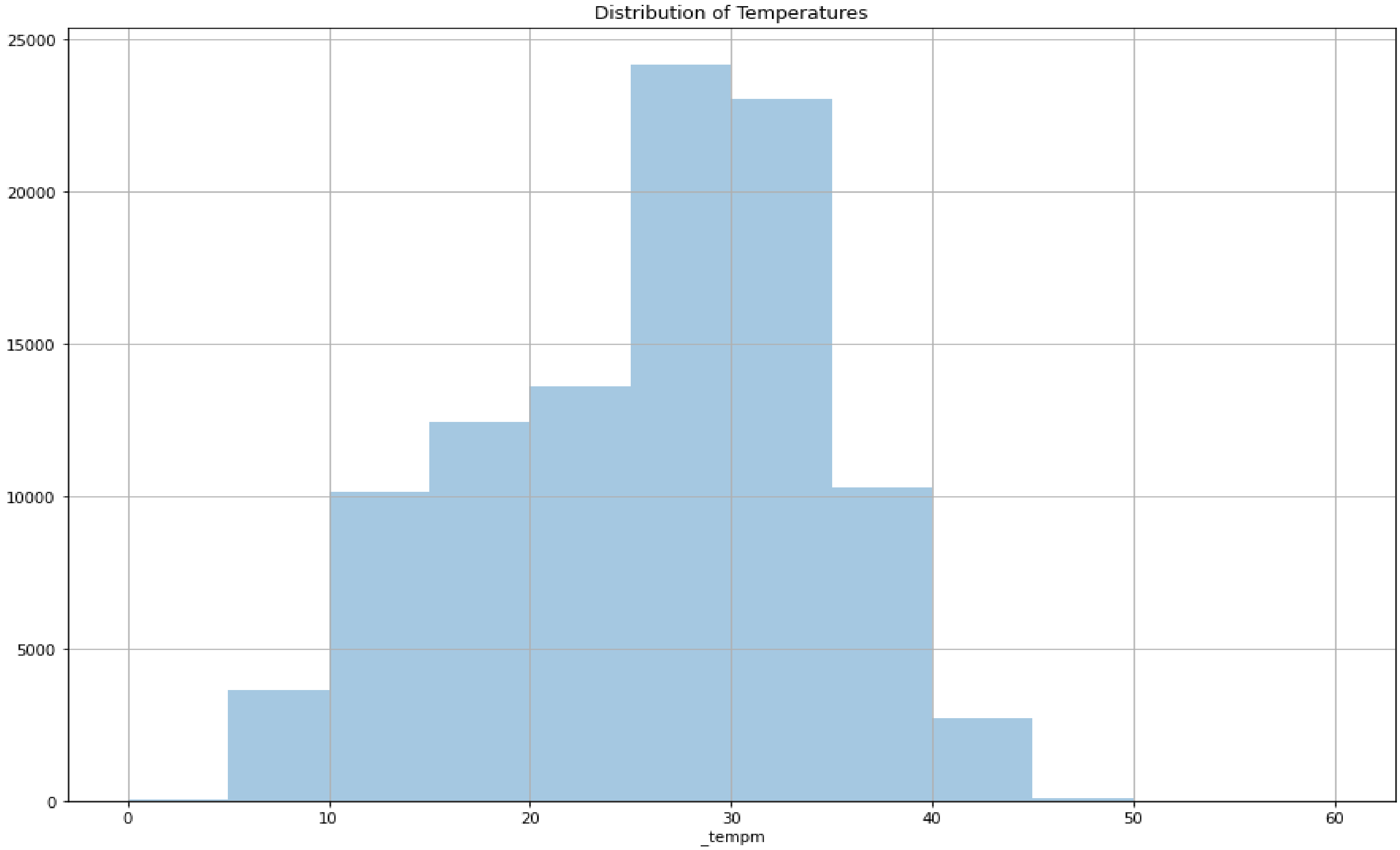
According to the above graph we can say that Haze and Smoke are the most common weather conditions in Delhi.

Another Bar Graph is for wind flow directions in Delhi:



North and West are the most common wind directions in the city.

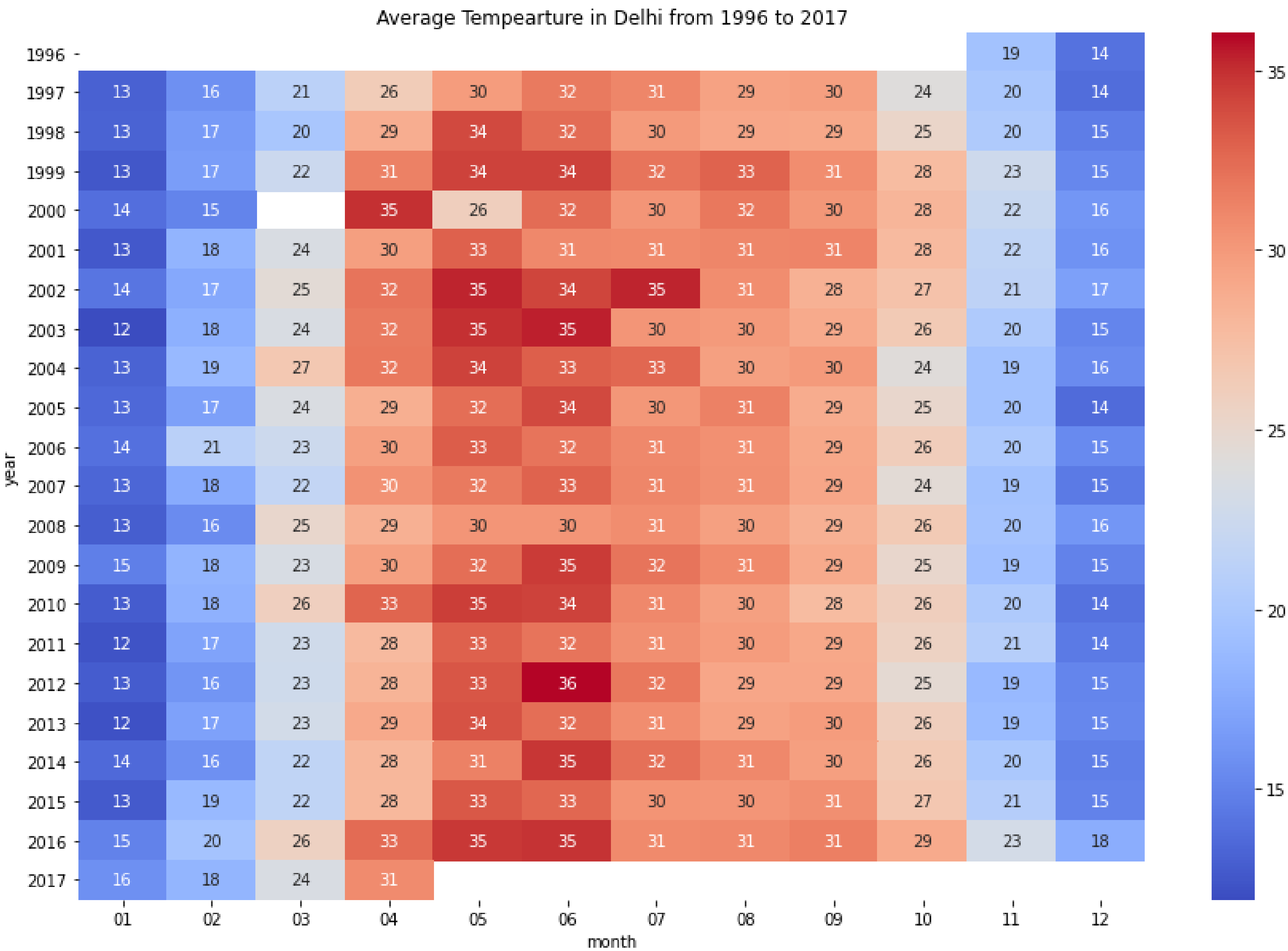
In weather analysis, it is very important to analyze the common range of temperature, for Delhi check the following Histogram:



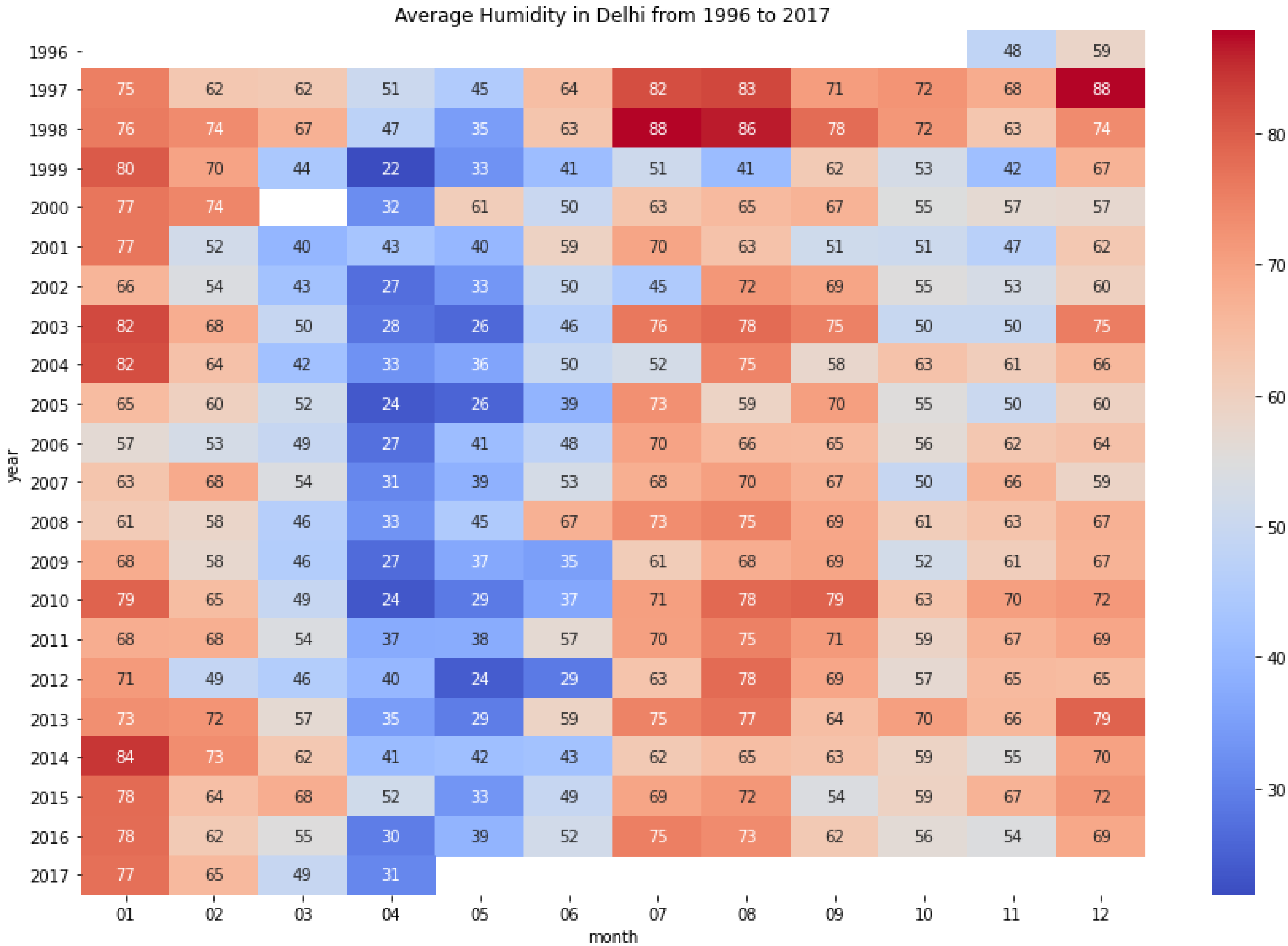
Most common temperatures in Delhi lies between 25 to 35 degrees.

We are using histogram for this because there are different temperatures in the graph from 1197 to 2017 of every day and we have to plot all the temperatures together which are in a range of 5 degrees.

Then we made some Heatmaps to check the temperature and humidity values for each month throughout the year.

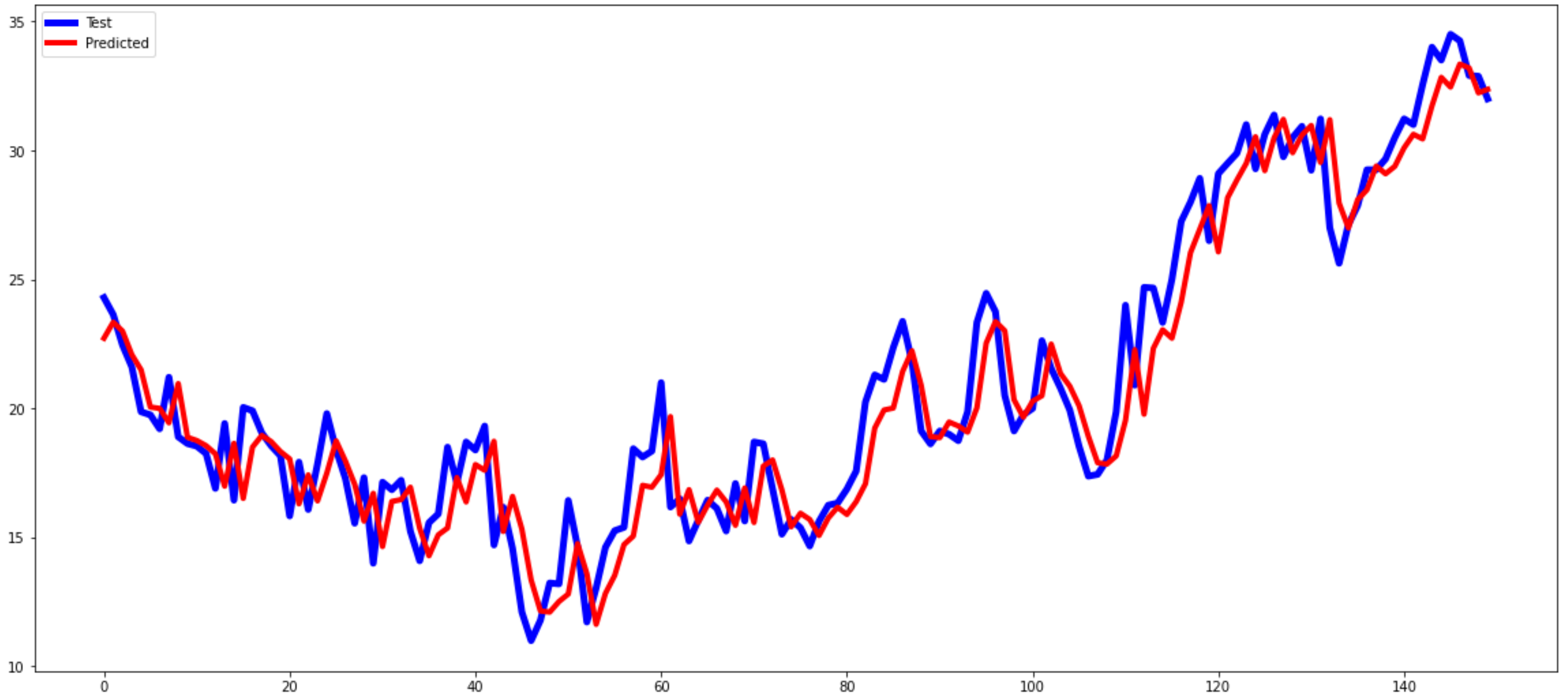


This is the Heatmap for the average temperature of each month of year from 1997 to 2017. From here we can analyze that the months January, February, March, October, November and December are relatively colder than other months.



This is the Heatmap for the average humidity of each month of year from 1997 to 2017. From here we can analyze that months March, April, May and June experiences less humidity.

Then we performed time series forecasting on the temperature column of the dataset using the above shown deep learning model. The result of our model is:



The blue plot is the actual plot of data given in the dataset and the red plot is of our predicted values by the model. The error between actual and predicted is very less.

The Mean Squared Error is our predicted values is 2.994

Conclusion

We concluded that the project like Stock Price Prediction and Weather Temperature prediction which contains time series data can be done easily with time series forecasting, the LSTM layers are really good to perform such prediction tasks.

REFERENCES

[1] [n.d.]. Weather Underground. https://www.wunderground.com/weather/api.

[2] 2018. Weather forecast using the sensor data from your IoT hub in Azure Machine Learning. https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-weather-forecast-machine-learning.

[3] DanBecker.2018. Using Categorical Data with One Hot Encoding. https://www.kaggle.com/dansbecker/using-categorical-data-with-one-hot-encoding/.

[4] JeffreyBurt.2017. Machine Learning Storms into Climate Research. https://www.nextplatform.com/2017/04/18/machine-learning-storms-climate-research/.

[5] Aditya Grover, Ashish Kapoor, and Eric Horvitz. 2015. A deep hybrid model for weather forecasting. In Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.ACM,379–386.

[6] Mark Holmstrom, Dylan Liu, and Christopher Vo. 2016. Machine Learning Applied to Weather Forecasting.

[7] Vladimir M Krasnopolsky and Michael S Fox-Rabinovitz.2006. Complex hybrid models combining deterministic and machine learning components for numerical climate modeling and weather prediction. NeuralNetworks19,2(2006),122–134.

[8] Federico Montori, Luca Bedogni, and Luciano Bononi. 2017. A Collaborative Internet of Things Architecture for Smart Cities and Environmental Monitoring. IEEE Internet of Things Journal (2017).

[9] Y Radhika and M Shashi. 2009. Atmospheric temperature prediction using support vector machines. International journal of computer theory and engineering 1,1 (2009),55. [10] Arne Sund. 2015. Using Amazon Machine Learning to Predict the Weather. https://arnesund.com/2015/05/31/ using-amazon-machine-learning-to-predict-the-weather/.

[11] Rohit Kumar Yadav and Ravi Khatri.2016. A Weather Forecasting Model using the Data Mining Technique. International Journal of Computer Applications 139, 14(2016).