📋

Final Year Research Paper

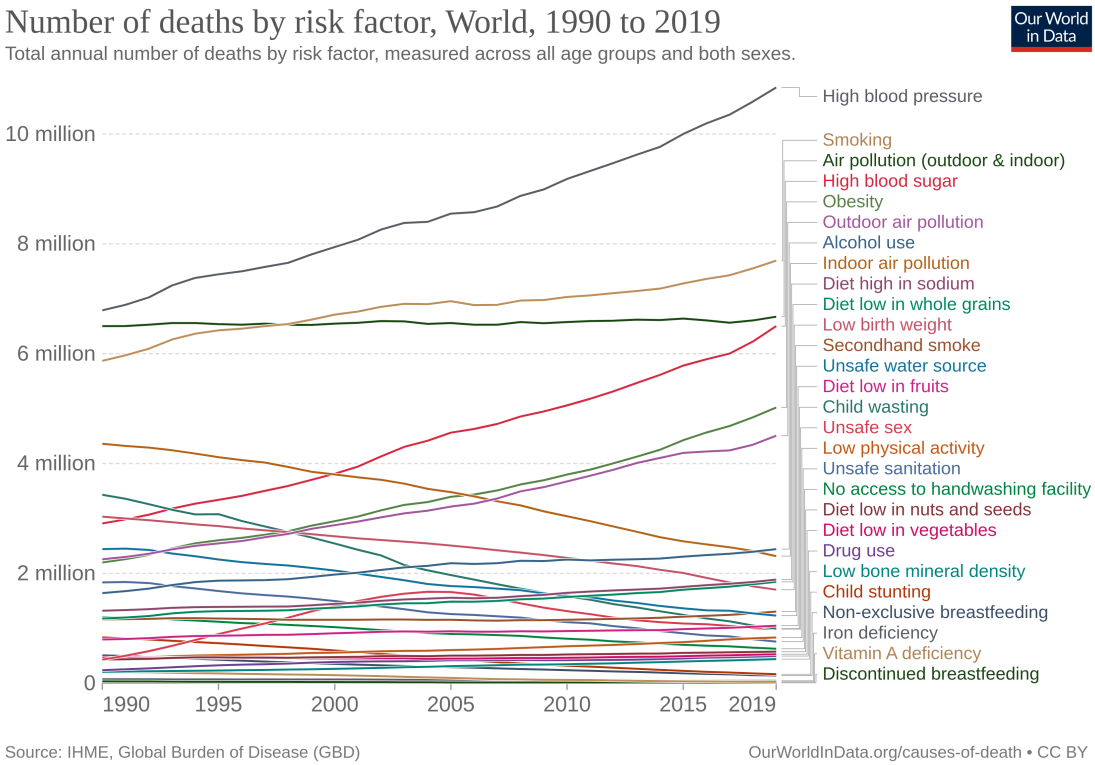
An online Air Pollution Forecasting system using LSTM and GRU

▼ Abstract

- With the development of the industry in the last few years. We are facing an issue related to air quality. We are not getting the proper air quality because of the pollution out there in the environment due to harmful gases from the industry. It will affect human health and it can cause a serious issues related to the lungs. Air pollution can cause by different ways in today's world like the CO2 released from the car and some harmful chemicals which are released in the air by the industry. Air pollution can spread with the flow of air means wind direction and speed. To overcome this real-time problem there is much research going on nowadays that can forecast air pollution but it required a lot of computational power. Our approach in this research is that we are going to forecast air pollution using the very famous deep learning technique Recurrent Neural Networks (RNN) based framework with special structure memory cell known as Long Short term memory (LSTM) and Gated Recurrent Unit (GRU). We can easily forecast air pollution using this easily by just providing some last day’s data to the model. It will forecast the next 24 hours of data by just providing the last 15 days of air pollution of data.

▼ Introduction

- According to the WHO (World Health Organization), air pollution is the contamination of the indoor or outdoor environment by any chemical, physical or biological agent that modified the natural characteristics of the atmosphere. Air pollution can be divided into 2 parts indoor pollution from households and outdoor pollution from vehicles and industry. Air pollution can be felt by Household combustion, motor vehicles, industrial facilities, and forest fires are common resources of air pollution. WHO data shows that almost all the global population (90%) breathes air that exceeds WHO guideline limits. Every 9 out of 10 people lives where air quality exceeds WHO guidelines. The World Health Organization (WHO) reported that air pollution causes 4.2 million premature deaths per year in cities and rural areas around the world. Air pollution in the cities and rural areas causes some dangerous diseases like stroke, heart disease, lung cancer, and acute and chronic respiratory diseases. Around the globe around 2.6 billion, people are exposed to dangerous levels of household air pollution. This is the data from WHO.
- As you can see in the world Air pollution leads the third largest cause of death[link].



- Air pollution forecasting techniques are being rapidly advanced and measuring pollution increase. Traditional approaches use some mathematical and statistical techniques[1]. This conventional forecasting model takes a lot of computational power to forecast the data. With recent advancements in technology, we come up with Deep Learning which is very good for solving real-time problems in various domains like computer vision, Natural Language Processing, and many more. With the promising results obtained by the Deep Learning model, we can adapt this to forecast air pollution[1].
- With the help of a deep learning approach, we can use the RNN (Recurrent Neural Networks) based framework which is LSTM (Logn short term memory) and Gated Recurrent Unit (GRU) with a special kind of memory cell attached to it. We have created a model with the help of the input layer (For providing the data to the model), 2 hidden layers (To process and learn about the pattern), and output layers (For output what the model has generated using the hidden layer). We can forecast the data with the help of providing the data of the last 15 days of data and we can forecast the next 24 hours' data which will be accurate according to the last 15 days of data.

▼ Literature Survey

- Deep Learning approaches have emerged as powerful solutions to mitigate these limitations over conventional methods [1]. The most popular Deep Learning techniques are Multi-Layer Perceptron (MLP), Deep Belief Network (DBN), Convolutional Neural Networks (CNN), Recurrent Neural Network (RNN), and Auto Encoder (AE). A particular RNN-based model for predicting air quality has drawn much attention in recent times [1].
- Many researchers are working on this problem of air pollution forecasting nowadays. Mostly they are focusing on the LSTM (Long short term memory) or GRU (Gated Recurrent Unit). But in this research, we are going to combine both of the famous models of Recurrent Networks[4]. In many fields, these 2 models are giving their best to provide the solution to the problem.

▼ Methodology

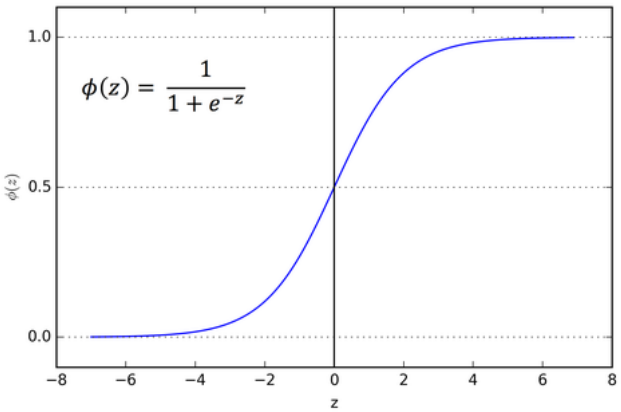
- In this paper, we are going to use the most popular framework of Deep Learning which is LSTM (Long Short term memory) and GRU (Gated Recurrent Unit). As we all know LSTM which has the special ability for storing the previous execution data and store in the memory and can be used for predicting data. Recurrent Neural networks suffer from short-term memory. If a sequence is long enough, they will have a hard time carrying information from earlier time steps to later ones.

- LSTM and GRU were created as the solution to short-term memory. They have internal mechanisms called gates that can regulate the flow of information. These gates can learn which data in a sequence is important to keep or throw away. The LSTM has a similar control flow as the recurrent neural network. It processes data passing on information as it propagates forwards. Every LSTM memory cell has the following Sigmoid and Tanh Activation function and three gates which are Input Gate, Forget Gate, and Output Gate.

$$i_t = \sigma(w_i[h_{t-1}, x_t] + b_i)$$
$$f_t = \sigma(w_f[h_{t-1}, x_t] + b_f)$$
$$o_t = \sigma(w_o[h_{t-1}, x_t] + b_o)$$

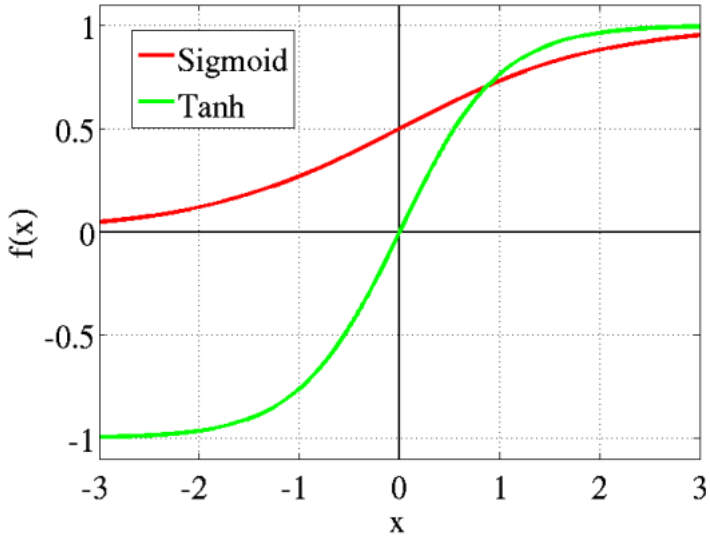
$$i_t \rightarrow represents\ input\ gate.$$
$$f_t \rightarrow represents\ forget\ gate.$$
$$o_t \rightarrow represents\ output\ gate.$$
$$\sigma \rightarrow represents\ sigmoid\ function.$$
$$w_x \rightarrow weight\ for\ the\ respective\ gate(x)\ neurons.$$
$$h_{t-1} \rightarrow output\ of\ the\ previous\ lstm\ block(at\ timestamp\ t-1).$$
$$x_t \rightarrow input\ at\ current\ timestamp.$$
$$b_x \rightarrow biases\ for\ the\ respective\ gates(x).$$

- The sigmoid activation function converts between 0 and 1. That is helpful to update or forget data because any number multiplied by ‘0’ will convert to 0, which can consider forgetting the data from the memory cell. And if the output is 1 then the value should be considered as the important data and kept that data in the memory cell. As we can see in the figure that the graph is from 0 to 1 and convert the values between 0 and 1.

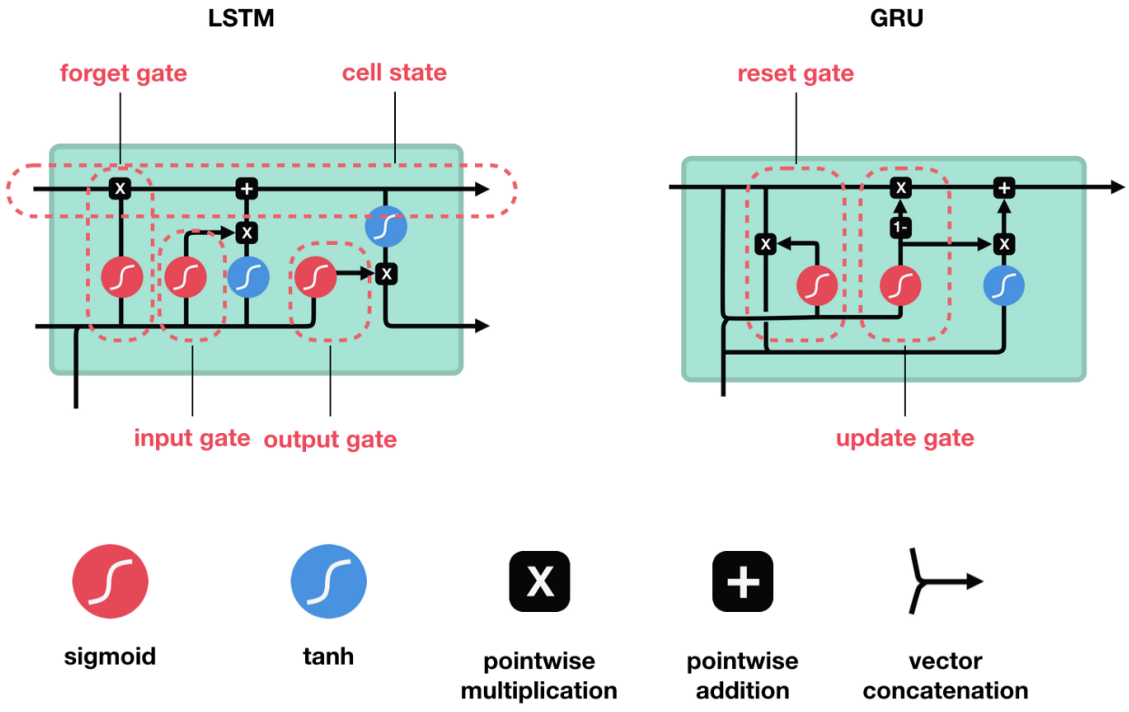


Tanh Activation Function Graph

- As similar to the sigmoid Activation function even tanh is also playing a major role in the LSTM memory cell. Tanh will squish the values between -1 and 1 and this is useful in the Input gate and convert the values between -1 and 1 and multiply with the existing data of that Input gate.



Sigmoid vs Tanh Activation function graph

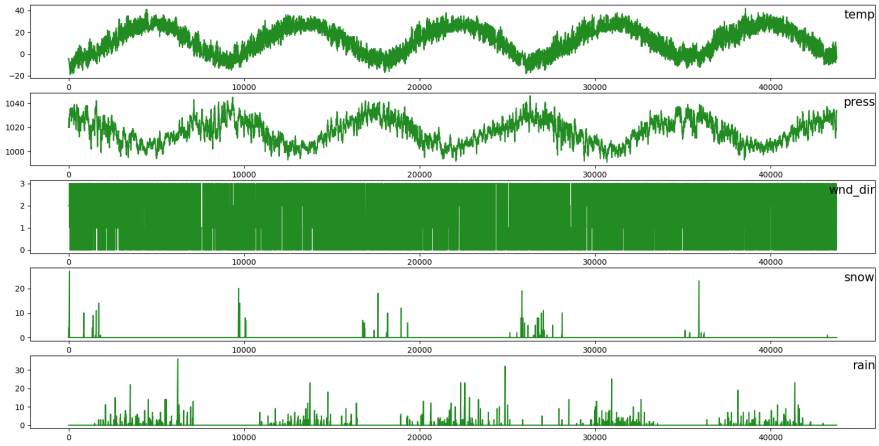


LSTM vs GRU

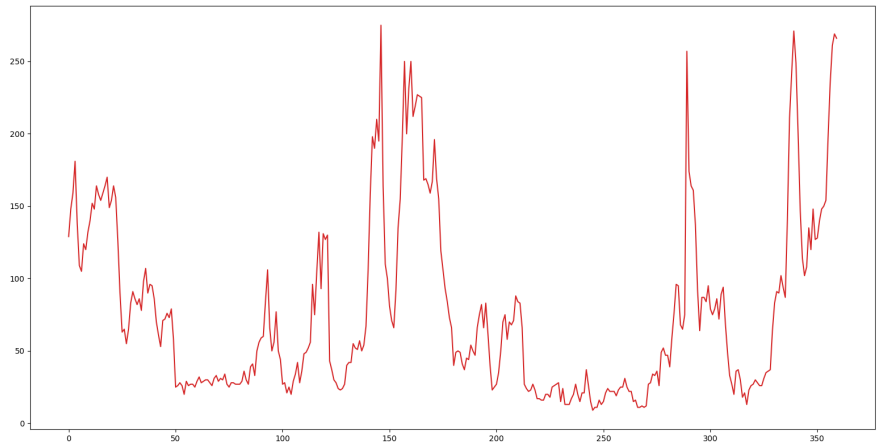
- As we can see in the figure that in LSTM we have 3 gates and in GRU we only have 2 gates. Both the LSTM and GRU are using the sigmoid activation function for the in there gate architecture. If the outcome for the sigmoid is 0 then forget the data and if the outcome is 1 then put the data and go further step. In Forget gate, it will decide whether the data which we are providing for the processing are important or not based on the output of the sigmoid function. Same for the GRU reset gate will define whether the cell state should reset or not based on the sigmoid activation function. If the outcome is 0 then reset the cell state or if 1 then keeps going on.
- We have used both the LSTM and GRU models for forecasting data. In our model, you have to give the last 15 days of data with the parameters of { date , pollution , dew , wind_dir , wind_spd , snow , rain , pollution } based on that we can forecast the next 24 hours of data which is pollution. We have to build a model to predict the next 24 hours of data. In this, the date is playing the most important role in forecasting data.
- For the implementation of this system we have to follow these steps :

1. Data Pre-processing
2. Creating of model
3. Saving of model
4. Generate output

- For Data Pre-processing firstly we have to understand the data which we have. To understand the data we can use the EDA. Firstly if in our data set there is a requirement for the Encoding of data we have to do that we can do that with the help of a Label Encoder or One Hot Encoder. After that, we have to normalize the data so our model can learn from that without getting more confused.

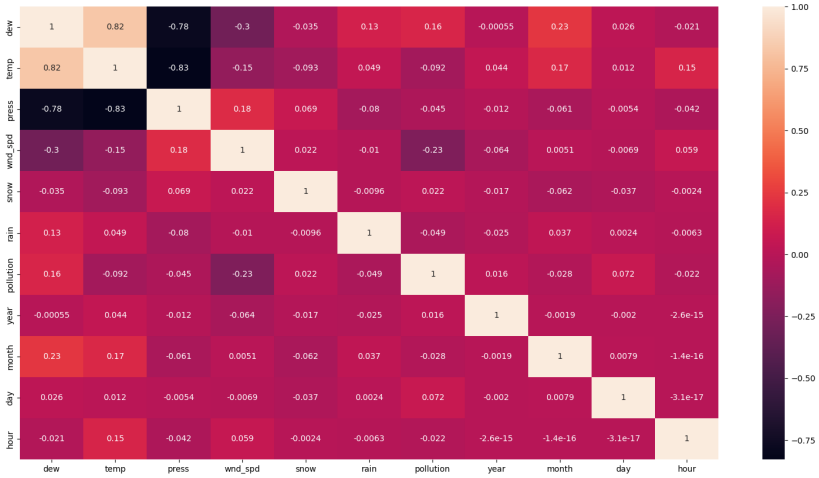


The Graph of input data to the model with fields of { temo , press , wind_dir , snow , rain }



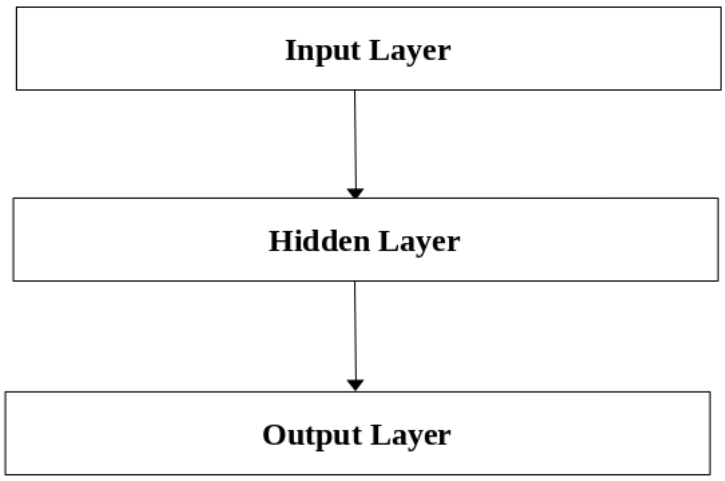
The Graphs of the pollution of the input data for the 360 hours which is 15 days.

- This is the graph of pollution from the last 15 days of data for every hour



The Cor-relation Matrix of the data.

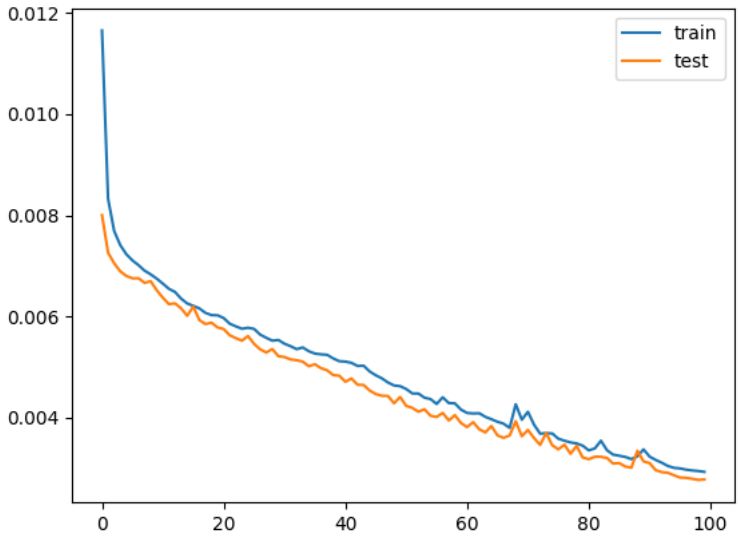
- For creating the model we have used the LSTM (Long short term memory) like the normal Deep learning architecture it has mainly three layers Input Layer, Hidden Layer, and Output Layer. In the input layer, we have created the input layer of the size of [360(no. of hours),11(no. of features)] with the activation function of relu. After that, we have added another LSTM layer of 50 neurons in it with another layer of Dense layer for the output layer and activation function of linear. For this, we have used loss functions of mse(Mean Standard Error), optimizer of Adam Optimizer, and metrics of accuracy. For training the model we have used the epoch of 100 and step_per_epoch is 25.



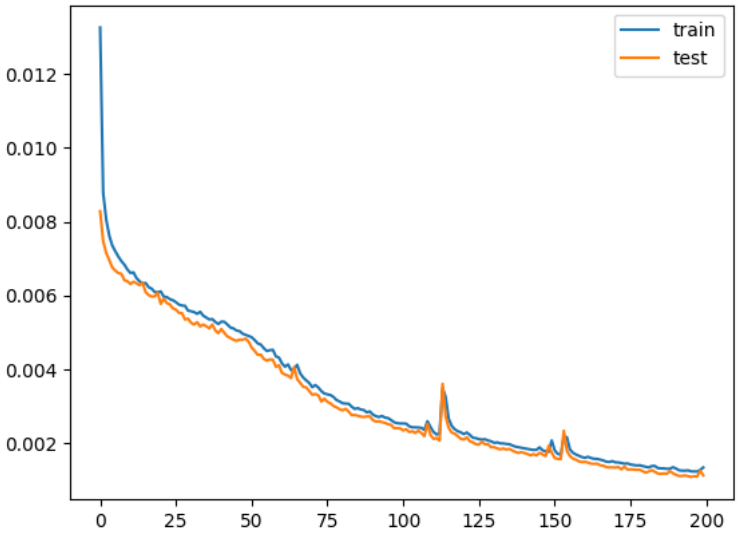
- After training the model we have to save that model because we cannot train the model again to predict the data. So once we train the model we will save it and we can use it for the further forecasting of data.
- To generate the output firstly we should have to give the last 15 days of data of each and every hour in total we have to give 360 hours of data to the model and it will predict the next 24 hours of data. It will generate the output of a single Dimensional array and we will generate a graph that will be easy to understand to us.

▼ Result

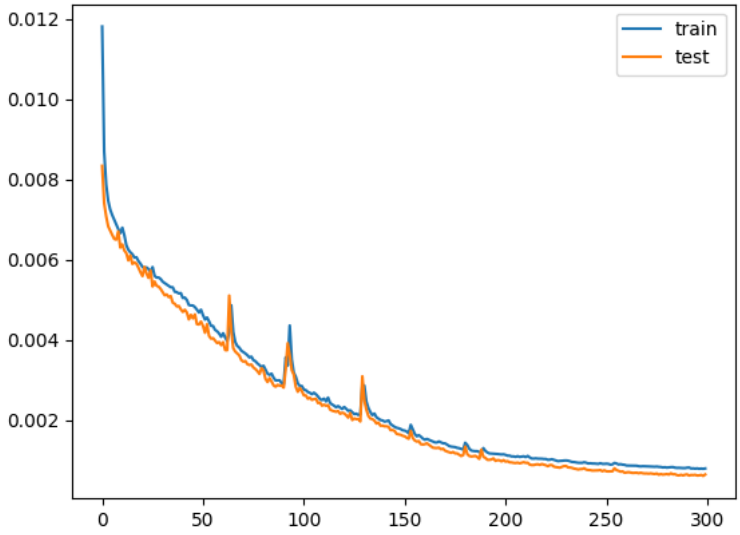
- We have trained the model with the 100 epoch, 200 epoch, and 300 epoch. We will plot the graph for all the three



100 Epoch graph with validation loss and loss

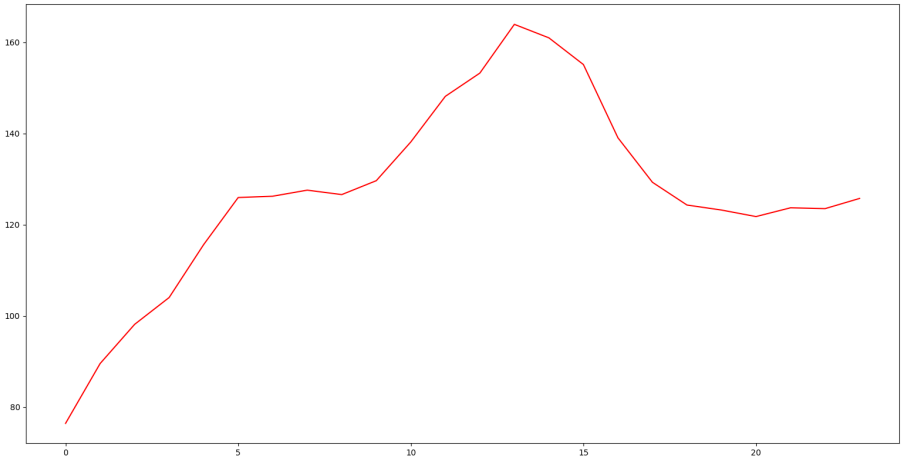


200 Epoch graph with Validation loss and loss



300 Epoch with Validation loss and loss

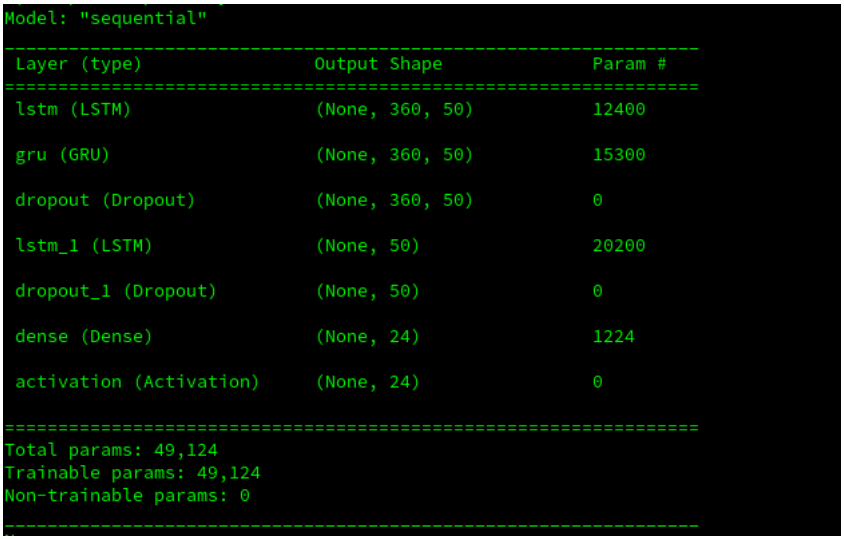
- As we can see from the above figure of all the three graphs, we can understand that in 200 epochs the model is performing good. So we will save that model and use this model for the forecasting of data.
- After giving the data to the trained model for the last 15 days it will generate the output for the next 24 hours.



The Graph of the output which has generated by the model.

- This is the model summary for our model, In this, we can see that we have

1. LSTM Layer
2. GRU Layer
3. Dropout Layer
4. LSTM Layer
5. Dropout Layer
6. Output Layer



▼ Conclusion

- The motive of this paper is to determine an efficient forecasting model for the hourly concentration level of air pollution. With the help of this model we can forecast the air pollution and we can see when the air pollution is going to be high. The result of work carried out supports the idea of a deep learning-based technique for forecasting air quality achieving promising performance. This work will take the input of the data for the last 15 days of data and can predict the next 24 hours of data. The result that the model has generated we can say that it is accurate based on the last 15 days of data but we cannot say anything about the environment.

▼ Reference

1. https://www.mecs-press.org/ijisa/ijisa-v11-n2/IJISA-V11-N2-3.pdf
2. https://www.researchgate.net/publication/340636552_Air_Quality_Forecasting_using_LSTM_RNN_and_Wireless_Sens
3. https://ieeexplore.ieee.org/abstract/document/8732985
4. https://link.springer.com/chapter/10.1007/978-981-15-3383-9_54
5. https://towardsdatascience.com/illustrated-guide-to-lstms-and-gru-s-a-step-by-step-explanation-44e9eb85bf21