MODELING OF AIR POLLUTION LEVELS USING STATISTICAL AND MACHINE LEARNING MODELS

MAJOR TECHNICAL PROJECT (DP 401P)

to be submitted by

GAGANDEEP TOMAR

for the

MID-SEMESTER EVALUATION

under the supervision of

Dr VARUN DUTT

SCHOOL OF COMPUTING AND ELECTRICAL ENGINEERING

INDIAN INSTITUTE OF TECHNOLOGY MANDI

KAMAND-175005, INDIA

SEPTEMBER, 2019

Contents

ABSTRACT

Air pollution cause massive damages to life with several pulmonary ailments. Thus, it is important to monitor air pollution levels in the atmosphere. Air pullution in India has become a serious issue and it has brought down life expectancy by 2.6 years[1]. In this project, students will be working on a data collected using IoT sensors in the Mandi district for predicting air pollution levels of the air pollutants $PM_{2.5}$, PM_{10} and the gaseous air pollutants NO_x, SO_2, CO and O_3 using statistical, Machine Learning and Deep Learning based time-series forecasting methods. The aim of the project is to produce such time series forecasting models which can predict the level of air pollutants given the past concentrations and the climatic indicators.

Keywords: *Add only IEEE keyword.*

Work done in I-phase

1.1 Objective and scope of the Work

The objective is to deliver such models which can predict the concentration of the air pollutants or the gaseous air pollutants given the past concentration of the pollutants. A prototype has already been installed in Mandi city which is already logging the concentration of these indicators. At the end of this project, we will deliver a model which can predict the concentration of these air pollutants with taking the data provided by the prototype installed in Mandi city as input.

As currently we don't have enough data available with us to train our models upon, we have taken Beijing $PM_{2.5}$ dataset from UCI Machine Learning Repository which contains $Pm_{2.5}$ data of 5 years logged on a hourly basis daily. Currently we will train our models on this dataset and as soon as we collect considerable amount of data from our installed prototye in Mandi city, we will start running our models from the data collected from the prototype and we will fine tune our models according to the data provided by the prototype.

1.2 Methodology

The project contains two major parts:

- Building a prototype which can collect the concentration of air pollutants in the atmosphere and log their concentration to a server.
- Building time series forecasting models which can predict the concentration of these air pollutants by taking the previous concentrations into account.

The first part is already completed by students under Dr. Varun Dutt and now the prototype has started logging data on to the server. Now, we are currently exploring deep learning models which can predict the concentration of the air pollutants logged by the prototype. As the deep learning models need extensive data and the data currently provided by the prototype is not enough, we have chosen a $PM_{2.5}$ concentration dataset provided by the UCI Machine Learning Repository in Beijing. This dataset contains the data of $PM_{2.5}$ concentration logged for over 5 years on a hourly basis daily. As now, the aim of the project is to build time series forecasting models which can predict the concentration of the air pollutants we are exploring regression techniques which

can predict the future concentrations of the pollutants based on the past history.
Following are the deep learning techniques/networks that we have tried to predict the $PM_{2.5}$ concentration from the Beijing dataset:

- MLP (Multilayer Perceptron)
- LSTM (Long Short-Term Memory) Network
- CNN (Convolutional Neural Network)

As time series data is data logged on different timestamps. To used it in a supervised macihne learning algoritm we had to transform this data.
A supervised machine learning algorithm needs data in the following form:

$$y = f(X) \tag{1.1}$$

On the other hand a time series consists data that is sampled at various timestamps. Hence timeseries can be easily realised as a function of time :

$$y = f(t) \tag{1.2}$$

where t equals to the timestamp at which the data of the quantity is needed.
But, in our application we need the data at the current timestamp to be a function of the data at the previous timestamps :

$$y_t = f(y_{t-1}, y_{t-2}, y_{t-3}...) \tag{1.3}$$

Hence for our application, we will use the observations at the previous time stamps or the lag-observations as an input to our models and our predicted observation would be the function of these lag observations. The numeber of lag-onservations used to predict the observation at the current timestep would become the lookback period for LSTM.

1.3 Results

We have tried three networks for now. Following are the details of the dataset:

- The dataset has 43000+ data points of $PM_{2.5}$ concentration.
- The train dataset was 80% of the total dataset and the test dataset was remaining 20%.
- The metric used is Mean Squared Error.
- The dataset was not normalised.

1.3.1 MLP

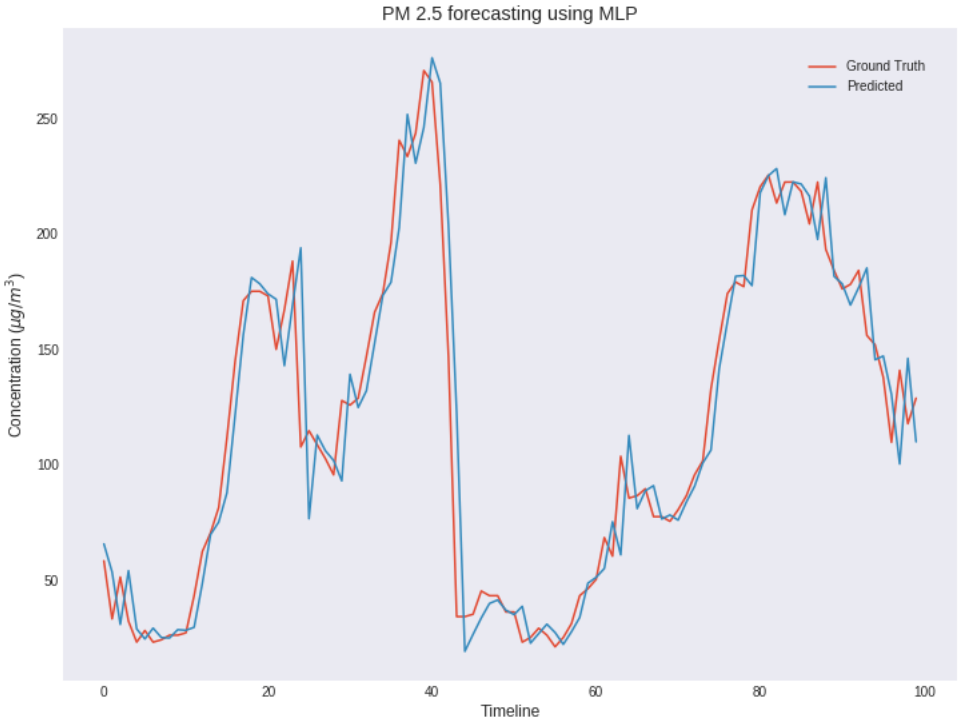

Figure 1.1: Predicted $PM_{2.5}$ concentration.

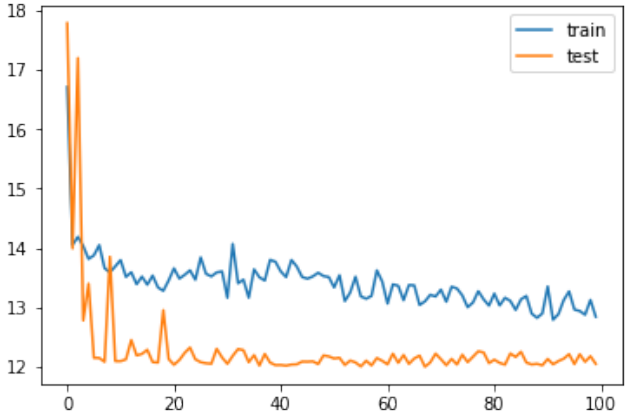

Figure 1.2: Loss curve for MLP.

Mean Squared Error : 533

1.3.2 LSTM

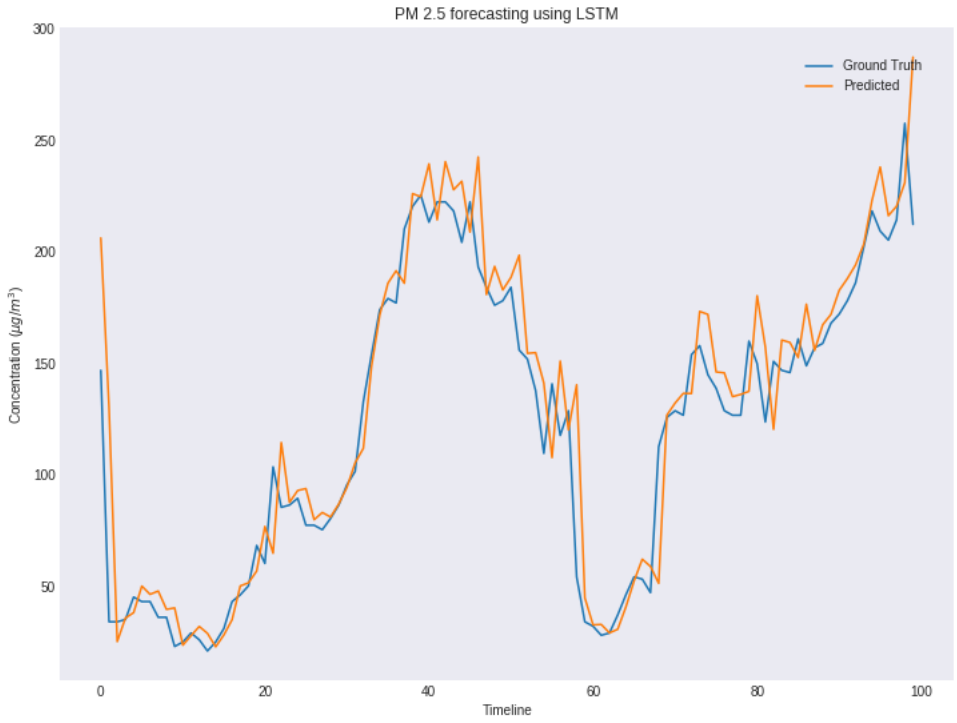

Figure 1.3: Predicted $PM_{2.5}$ concentration.

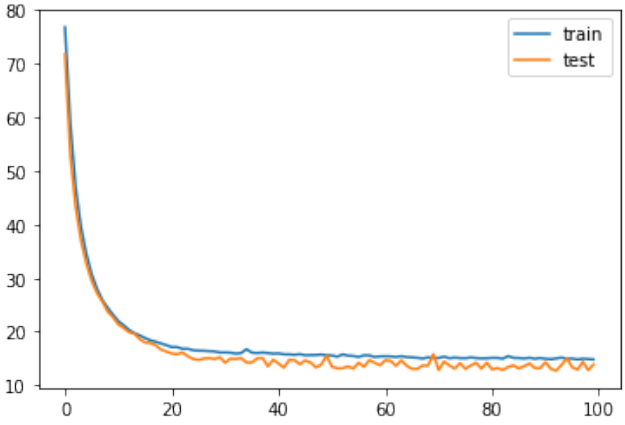

Figure 1.4: Loss curve for LSTM.

Mean Squared Error : 640

1.3.3 CNN

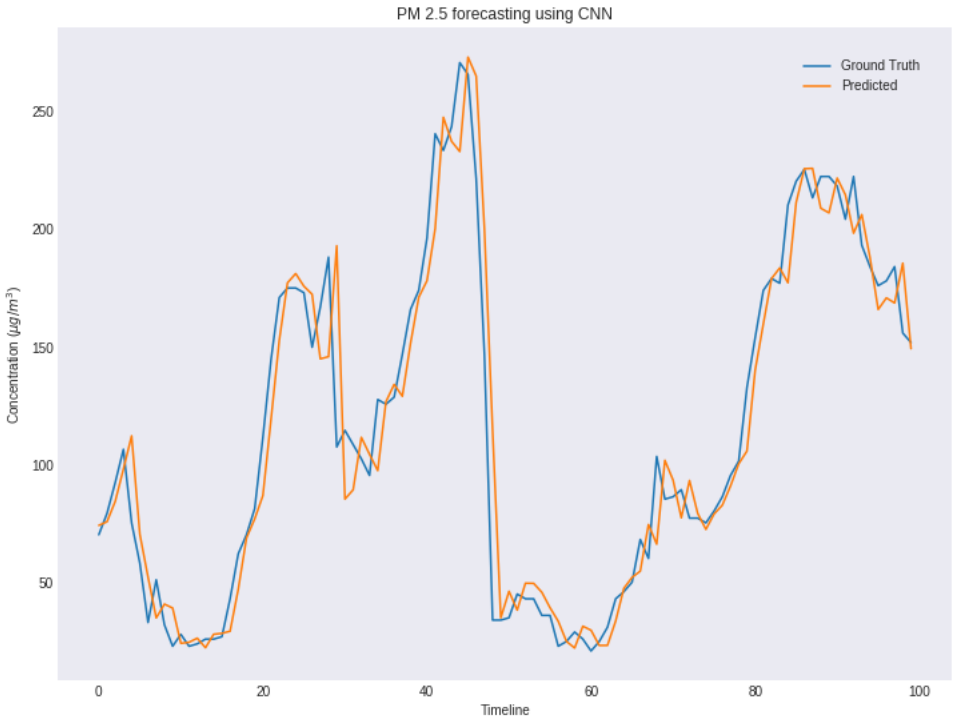

Figure 1.5: Predicted $PM_{2.5}$ concentration.

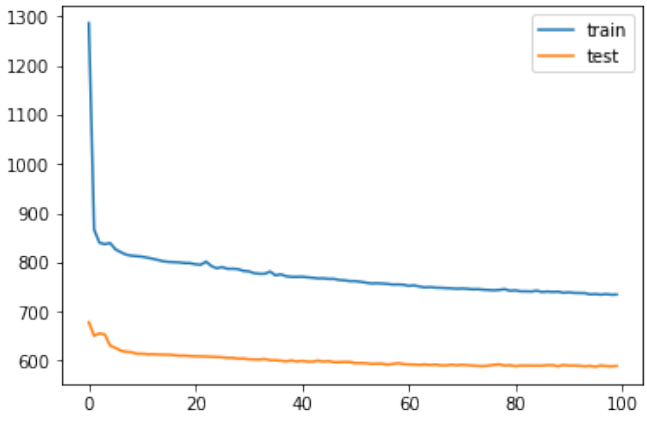

Figure 1.6: Loss curve for CNN.

Mean Squared Error : 589

1.3.4 Grid search for MLP and LSTM

The μ came out to be 0.8185 for the grid search (for MLP).

1.4 Conclusion and Future Work

Currently we have just tried three models without any fine tuning. We plan on exploring more methodologies for time series forecasting and we finally plan on building an ensemble model with the models which will be performing the best. As an ensemble model would require adding on some hyper-parameters for which we might add a network to make it learn the hyper parameters or we might try attention based learning.
After exploring all the approaches in the literature, the project will focus on theoretical models of forecasting which involves borrowing ideas from high dimensional probability theory, stochastic processes and stochastic calculus. These approaches will prove benificial in coming up with more domain specific objectives for training deep learning models.