DeepAir: Air Quality Prediction using Deep Neural Network

Pratyush Singh
Department of Electrical Engineering
Indian Institute of Technology Palakkad

Lakshmi Narasimhan T
Department of Electrical Engineering
Indian Institute of Technology Palakkad

Chandra Shekar Lakshminarayanan
Department of Computer Science and Engineering
Indian Institute of Technology Palakkad

Abstract— **Air pollution is a common cause for several major health hazards and climate change. Air quality predictors are required to plan human activities in a given geographic locality and minimize the negative effects of pollution. We look at the problem of predicting a sequence of future pollutant concentrations. We propose a solution based on Long Short-Term Memory (LSTM) networks, which are deep neural networks that are known to perform well on sequential prediction problems. We conduct adequate number of experiments to select the best possible features and perform hyperparameter optimization using the Python package Talos. We employ the root mean square error (RMSE) as our performance measure. The presented predictor achieves an RMSE of less than 1.2 for the pollution dataset obtained from Indira Gandhi International Airport at New Delhi, India. To the best of our knowledge, this is the highest level of prediction accuracy reported, so far in the literature, for this dataset.**

Keywords— LSTM, Air Quality, Time Series Data, $PM_{2.5}$, Predictive modeling, TALOS

I. Introduction

Air pollution is a major risk factor for heart disease, stroke, chronic obstructive pulmonary disease, lung cancer, acute respiratory infections and exacerbates asthma. Prediction of air pollution is important for better management of public health given the alarming rate of increase of $PM_{2.5}$ (particulate matter with a diameter less than 2.5 micrometres – $PM_{2.5}$, can enter the human respiratory system easily). Studies show that the level of pollution also depends on meteorological factors such as wind speed, temperature, pressure and wind direction. Further, the pollutant concentration has inherent temporal dependence wherein the future sequence of pollutant levels is dependent on the past sequence of pollutant levels, meteorological factors, and traffic conditions. Long-Short Term Memory (LSTM) networks are deep neural networks that have been known to perform well in sequential prediction tasks. In this paper, we propose an LSTM based solution to predict future air pollutant concentrations.

A. Motivation

It is important to know in advance the trend of the pollution levels so that necessary precautionary measures can be taken to prevent oneself from the harmful effects of long-term exposure to pollution. Thus, the prediction of pollutant levels is one of the most important problems for civil agencies and is also important in weather monitoring.

According to WHO data [1], in India, an estimated 1.5 million people died from the effects of air pollution in 2012. At least 140 million people breathe air of quality that is 10 times or more above the WHO safety limit; thirteen of the world's 20 cities with the highest annual levels of air pollution are in India. New Delhi is among the most polluted cities in India as well as in the world. Air pollution contributes to the premature deaths of about 2 million Indians every year [2]. The common air pollutants present are particulate matters ($PM_{2.5}$ and PM_{10}), carbon monoxide (CO), ozone (O_3), nitrogen dioxide (NO_2) and lead (Pb). The Government of India has installed sensors in many cities and these sensors can detect the concentration of these pollutants at the specific geographic areas. These sensor measurement data is made publicly available by the central pollution control board (CPCB).

Of the several aforementioned pollutants, Particulate Matter (PM) 2.5 is of interest: their diameter is less than 2.5 micrometres and can penetrate deep into the bronchioles of the human respiratory system. Thus, predicting the level of $PM_{2.5}$ is a key task is determining air quality. In this paper, we consider the dataset obtained from the Indira Gandhi International Airport region, New Delhi. We use this dataset to train an LSTM network to predict the future sequence of $PM_{2.5}$ concentrations in this region.

B. Organization of the Paper

This section provides an overview of the contents in this paper. A review of the current literature that is relevant to the air pollution prediction problem is given in Section II. The state-of-the-art methods used for air quality prediction are described in Section II. In Section III, we describe the dataset by mentioning the time-period for which the data was obtained (by CPCB), the repositories from which these datasets can be downloaded along with the format in which they are available.

978-1-7281-1895-6/19/$31.00 ©2019 IEEE

In Section IV, we discuss the importance of the feature selection and describe the method of correlation used in our experiments to determine the best features for training our deep neural network. The architecture of the LSTM model is explained in Section V along with the data pre-processing methods and libraries. We also discuss the TALOS implementation for optimizing the hyperparameters of the neural network. Section VI discusses the novelty, performance and advantages of our predictor, and a comparison is made with the existing methods described in Section II. Finally, in Section VII, we provide the conclusion and possible directions for future research in this area.

II. Literature Review and Related Work

In this section, we discuss the most relevant and state-of-the-art solutions proposed in the literature that employs deep learning methods for air quality prediction.

Chaudhary et al [3] proposed a deep learning based stacked LSTM model to predict future air pollutant's concentration. They consider features such as meteorological factors, pollutants (NO_2, CO, $PM_{2.5}$ etc.), occurance of festivals/national holidays and traffic information. From the data given at a time instant, they predicted pollutant's concentration for the next 1 hour, next 6 hours and the next 12 hours, for localities in Agra and Delhi.

Reddy et al [4] use LSTM model for forecasting pollution in the Beijing area. They predicted $PM_{2.5}$ concentration for just the next time step. They consider only the meteorological factors as features for air quality prediction.

Mahajan et al [5] propose a neural network autoregression (NNAR) method for the prediction of $PM_{2.5}$ concentration. They present a prediction model for predicting PM2.5 for the next one hour only. The paper also provides a comparative analysis of the prediction performance for additive version of the Holt-Winters method, autoregressive integrated moving average (ARIMA) model and NNAR model.

Zheng et al [6] employ meteorological data, traffic flow, human mobility and points of interest (POIs) as some of the features to predict air quality at a given location. They have implemented a co-training based semi supervised learning approach that consists of two separate classifiers, artificial neural network (ANN) based spatial classifier and conditional random field (CRF) based temporal classifier.

Perez et al [7] employ a three layer neural network to predict $PM_{2.5}$ concentration and use previous 24 hours of data for prediction. Roy [8] describe a method to predict PM10 concentration using an ANN based approach. They employ a multilayer feed forward network for prediction using meteorological data.

In this paper, we propose a deep neural network to predict $PM_{2.5}$ concentration for the next 23 hours from a given time instant. We use a non-stacked LSTM model that is trained with the data on the pollutant concentration and other key features at previous time steps. In our work, we have developed a single layer LSTM network which can retain a longer memory to predict better results. Further, we choose only the required features to mitigate over-fitting and improve prediction performance. Thus, the proposed deep neural network has low computational complexity and better performance.

III. Datasets

For training and testing purposes, we compiled the available real-time datasets for pollutant concentration, meteorological factors and traffic data at Indira Gandhi International Airport from various sources.

The following real-time data for every hour was collected from 1st January 2018 to 30th September 2018:

- Meteorological Factors: Temperature, pressure, humidity, wind speed, wind gust, wind direction, dew point, precipitation, condition (cloud cover). The Accuweather website provides this data in javascript object notation (JSON) format.
- Pollutants Concentration: $PM_{2.5}$, PM10, NO2, NOx, NO, CO, Ozone. This data is fetched hourly for CPCB website, where it is available in XML format.
- Traffic Information: This data was collected in terms of the average speed of the vehicles at each hour. Corresponding traffic information at the sensor location is captured using "Here" maps traffic APIs. This data is available in XML or JSON format, which includes information on speed and congestion for the region(s) defined in each request.

The sources of the dataset are shown in Table I.

TABLE I. Sources of the Datasets

Data	Source
Pollutant Data	https://app.cpcbccr.com/ccr/#/caaqm-dashboard-all/caaqm-landing
Meteorological Data	https://developer.accuweather.com/
Traffic Data	HERE maps Traffic API

This dataset contains large volumes of measurements of several correlated factors. Our main motive is to get the best prediction results by using minimal number of features. Thus, we need to choose the best set of features that succinctly captures all the required correlations for predicting $PM_{2.5}$ levels without resulting in any overfitting of the data.

IV. Feature Selection

The necessity of selecting the best features for the deep neural network model is a key step in minimizing overfitting and improving accuracy. Through feature selection, we select the features which contribute most to the prediction accuracy. Irrelevant or partially relevant features can negatively impact the performance of prediction. The following are the advantages of performing this feature selection operation:

- It reduces overfitting. That is, it minimizes the chance of making decisions based on noisy data and/or outliers.
- It improves accuracy as there are less ambiguous and noisy data.
- It reduces training time and computational complexity.

We propose the following correlation analysis technique for performing this feature selection.

A. Correlation Analysis

Correlation coefficient helps in finding the statistical relationship between two random variables, it can take values in the range (-1, +1), where +1 indicates the strongest possible agreement and -1 the strongest possible disagreement.

In this paper, we have used the Pearson correlation coefficient and Spearman correlation coefficient to analyze the dependencies between each of the features with $PM_{2.5}$. Pearson's correlation coefficient is the ratio of the covariance of the two features and the product of their standard deviations. It is a measure of the linear correlation between two random variables. The Spearman correlation coefficient is defined as the Pearson correlation coefficient between the rank variables. It assesses how well the relationship between two variables can be described using a monotonic function.

We illustrate this correlation analysis in figures 1, 2 and 3. The plots in Fig. 1, Fig. 2 and Fig. 3 are obtained by computing the moving average of the correlation coefficients across the time series data. It can be inferred from Fig. 1 that the temperature and $PM_{2.5}$ have a positive correlation which can mean that, when the temperature is high, the concentration of $PM_{2.5}$ is also high. Fig. 2 shows a negative correlation between humidity and $PM_{2.5}$; whereas, Fig. 3 does not provide a clear relationship between the levels of CO and $PM_{2.5}$, hence, it is inconclusive to decide whether there is a positive or negative correlation between CO and $PM_{2.5}$.

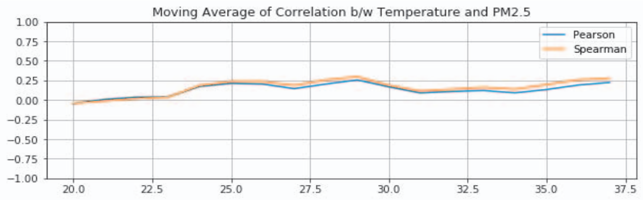

Fig. 1. Moving average of correlation b/w temperature and $PM_{2.5}$

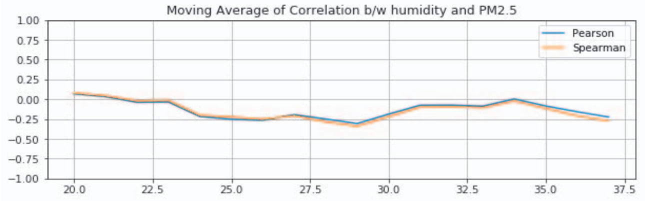

Fig. 2. Moving average of correlation b/w humidity and $PM_{2.5}$

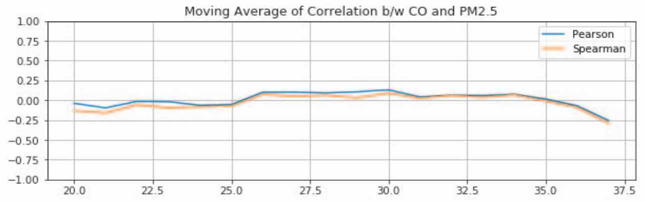

Fig. 3. Moving average of correlation b/w CO and $PM_{2.5}$

The emission sources of the pollutants CO, NO, NO_2, NO_x, Ozone, PM are quite different as shown by CPCB [9]. Relative humidity and dew point are indeed concerned with the amount of water vapour in the air, but there are differences. Dew point is of great interest to meteorologists, because it is a basic measure of the state of the atmosphere in terms of how much water vapour is present. Unlike relative humidity, which depends on both temperature and the amount of water vapour (so that the relative humidity can change even if the amount of water vapour remains the same). For this reason, we concern ourselves to consider the dew point for training purposes. The pressure and average speed of vehicles quantity have a slight positive correlation and needs to be taken into account. We also perform the correlation analysis with several other features available in the dataset. Some of the inferences from this correlation analysis are summarised in Table II.

From this analysis, we conclude that the following are the key training features to be considered for high accuracy of air quality prediction: temperature, pressure, wind speed, dew point, precipitation, average speed of vehicles and wind direction (as a categorical feature).

TABLE II. CORRELATION INFERENCES

Correlation b/w a feature and $PM_{2.5}$	**Comment(s)**
Temperature	Positive correlation
Pressure	Slight positive correlation
Wind speed	Slight negative correlation
Dew point	Slight negative correlation
Humidity	Negative correlation
CO	Inconclusive
NO	Positive correlation
NO2	Positive correlation
NOx	Positive correlation
Ozone	Slight positive correlation
PM10	Highly positive correlation
Precipitation	Slight negative correlation
Average speed of vehicles	Slight positive correlation

B. Data Pre-processing

The deep neural network receives data in chronological order for which the initial step is to arrange all the input data (meteorological, pollutant and average speed) in a time series format. The dataset is converted into float type by using the MinMaxScaler library of Python's sklearn package. Each value of the data vary between 0 and 1. The categorical feature, i.e., wind direction is encoded using sklearn library. Finally, the data is reshaped into a 3D format as required by the architecture of the deep neural network that is used.

In the next section, we describe the architecture of the deep neural network that is designed for prediction of air quality using the above mentioned features.

V. Architecture

We employ a deep neural network that is constructed using the long short-term memory (LSTM) model, which is a specific type of recurrent neural network (RNN), to predict future values of the air pollutant ($PM_{2.5}$) concentration. The LSTM model can retain memory from relevant past events for better prediction. We formulate the problem of predicting pollutant concentration as a time series based problem, where the current pollutant level is dependent on the previous pollutant level, meteorological factors, and traffic information. Therefore, to predict the pollutant concentration at time *t+1*, we not only take the current input data at time *t,* but also use the predicted values at time *t-1, t-2, ..., t-N,* where *N* is the memory length. The basic LSTM unit consists of an input gate, an output gate and a forget gate. This cell remembers values over time and these gates regulate the flow of information in each LSTM unit. Our LSTM layer has 75 units. Weights in each unit are available in input gate, output gate, forget gate and cell state. The best results are obtained when the model has three layers: input layer, hidden layer (LSTM), output layer. The hidden layer has 75 LSTM units/neurons. The input layer is used to provide the pre-processed data to the LSTM model.

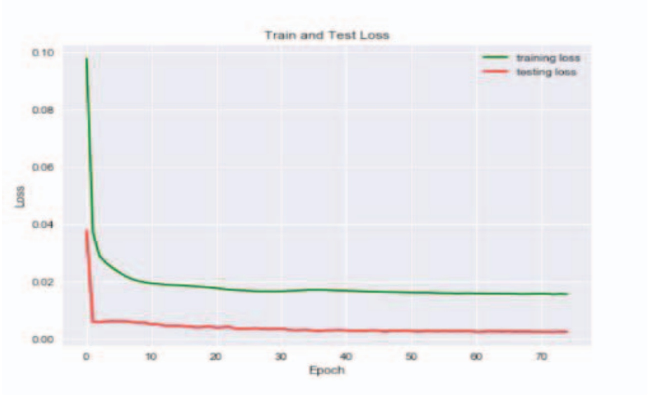

Fig. 4. Validation loss curve.

A. Hyperparameter optimization

Talos [10] is a Python package that allows us to configure, perform and evaluate hyperparameter optimization for neural networks. This optimization can result in better accuracy across a wide range of prediction tasks. Talos provides a hyperparameter optimization method using the package Keras. In our experiment, we use Talos to fine tune many hyperparameters such as batch size (108), epochs (75), dropout (37.3%), optimizer (Nesterov Adam optimizer), activation function (Sigmoid).

VI. Experiment and Results

In this section, we present the experimental results obtained using the proposed deep neural network. Root mean square error (RMSE) is the metric that we employ to compare the results for our experiments. RMSE can be defined as

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(Predicted_i - Actual_i)} .$$

Mean absolute error (MAE) is used to evaluate the training and testing loss. We compare our results with the real-time pollutants concentration at Indira Gandhi international airport region, New Delhi. This data is available from the CPCB website which is provided by the Government of India. The training is performed on more than 8 months of data. We predict the $PM_{2.5}$ concentration for each of the next *23 hours* from a given time instant (as opposed to just the next hour or the sixth hour as done in other methods discussed in Section II). The RMSE achieved in this process is less than 1.2 for all these predictions. We compare our results with other state-of-the-art methods in Table IV. The validation loss achieved for our model is as low as 0.002, this is shown in Fig. 4. In comparison, [3] reported the RMSE for the Delhi region as 3.183 for the next 1 hour, 12.37 for the next 6 hours and 17.58 for the next 12 hours, while it is 1.58 in [5]. Thus, the proposed method produces air quality predictions with the best known level of accuracy.

We also present our results of the stacked LSTM model with different number of LSTM layers in Table III. The RMSE achieved in both the cases is low. However, it increases the training time and algorithm computational complexity; the unstacked model still has the best accuracy. We conclude that this is a result of the over-fitting caused by the redundant units in the neural network and outliers in the noisy data.

TABLE III. Stacked LSTM Model Results

No. of LSTM layers	Neurons in 2nd LSTM Layer	RMSE
2	50	1.9
3	50 (2nd and 3rd Hidden layer)	1.5

TABLE IV. COMPARISON OF RMSE OF MODELS (NEXT HOUR PREDICTION)

Model	RMSE
Current work	**1.2**
Stacked LSTM [3]	2.513 (for Agra) 3.183 (for Delhi)
NNAR Model [5]	1.58

VII. CONCLUSION AND FUTURE WORK

New Delhi, India, is one of the most populated cities in the world; predicting air quality can help in making several civil decisions and maintaining the health of the citizens. In this paper, we proposed a deep neural network architecture to predict the concentration of $PM_{2.5}$. We identified the best features for training purposes from the real-time dataset using correlation coefficients for the chosen geographical location. We conducted several experiments using the data obtained from CPCB at the Indira Gandhi International Airport Area. The RMSE achieved from the proposed deep neural network is less than 1.2 for each of the next 23 hours of prediction from a given time instant. Future extensions of this work include developing an online LSTM algorithm for $PM_{2.5}$ concentration prediction and to predict pollutants concentration at locations without direct sensor measurements.

REFERENCES

[1] “India takes steps to curb air pollution,” *Bulletin of the World Health Organization*, vol. 94, no.7, pp. 481-556, July 2016.

[2] “Air pollution in india — Wikipedia, the free encyclopedia,” [Online] Available: https://en.wikipedia.org/, 2019.

[3] V. Chaudhary, A. Deshbhratar, V. Kumar, D. Paul, and Samsung, “Time series based LSTM model to predict air pollutants concentration for prominent cities in India,” 2018.

[4] V. N. Reddy, P Yedavalli, S Mohanty and U Nakhat, “Deep air: Forecasting air pollution in Beijing, China,” 2017.

[5] S. Mahajan, L. Chen and T. Tsai, “An empirical study of PM2.5 forecasting using neural network,” *IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computed, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation*, San Francisco, 2017.

[6] Y. Zheng, F. Liu, and H.-P. Hsieh, “U-air: When urban air quality inference meets big data,” 19th *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, New York, 2013.

[7] P. Perez and J. Reyes, “Prediction of particulate air pollution using neural techniques,” *Neural Computing & Applications*, vol. 10, no. 2, pp. 165 – 171, May 2001.

[8] S. Roy, “Prediction of particulate matter concentrations using artificial neural network,” *Resources and Environment*, vol. 2, pp. 30–36, March 2012.

[9] Central pollution Control Board, “ENVIS Centre on Control of Pollution Water, Air and Noise,” [Online] Available: https://cpcbenvis.nic.in/envis_newsletter/Air%20pollution%20in%20Delhi.pdf

[10] Mikko Kotila, “Hyperparameter optimization for keras models,” [Online] Available: https://github.com/autonomio/talos