Pollution Estimation and Prediction of Pittsburgh

Haineng Huang hainengh@andrew
Jianfeng Su jianfens@andrew
Wenyan Zheng wenyanzh@andrew
Kaixin Song kaixins@andrew

Abstract

In modern society, air pollution exerts a critical negative impact on the environment and the health of the human population. The Air Quality Index (AQI), is a dimensionless index that quantitatively describes the air quality condition, and the higher its value, the worse the air quality condition and the greater the negative impact on human health, ecology, and the environment. We use data from U.S. Environmental Protection Agency and macro-trends to predict AQI values in Pittsburgh through $PM_{2.5}$, PM_{10}, SO_2, CO, and O_3 concentrations. The mean squared error (MSE) is applied to the experiments in this project to assess the overall performance of the algorithm. To estimate and predict linear AQI regression in Pittsburgh, two simple model, Linear Regression and Deep neural network,are applied to the AQI from the forecasting system. Then we try to take the other pollution sensors' information in to consideration to better predict the AQI of Pittsburgh, so we use the Convolutional Neural Networks (CNN) on filtered 3 days ozone data transformed to a 2D image to predict the Ozone Value of Pittsburgh, Which has a better performance than normal models only using Pittsburgh's Ozone data. It is found that interregional pollutant transportation has a great impact on the air quality of Pittsburgh.

1 Introduction

Poor air quality is one of the greatest public health threats of our time. According to the World Health Organization, more than 90% pollution, which of the world breathes unhealthy air pollution has become the fourth largest risk factor for human health on the planet. Past studies indicate that globally, 8.8 million people die indirectly each year because of poor air quality, and 1 in 8 premature deaths in the world are due to air pollution. The AQI, or Air Quality Index, measures air quality over a given period of time for a given area. The value of the AQI between 1 and 50 is considered satisfactory, and air pollution poses little or no risk. Factors affecting AQI etc. values include the level of ground-level ozone, particle pollution, nitrogen dioxide, carbon monoxide, and sulfur dioxide, etc in the air. Due to the industrialization and urbanization of Pittsburgh, in our study, we focus on the prediction of AQI Value, which generally represent the air quality (and the CNN part is concentrated on Ozone, O3) in the region and investigate the correlation between pollution parameters and AQI values in the control. This study uses air quality data from Pittsburgh and sensors all across the United States ,and uses data engineering and machine learning methods to process and predict air pollutants.

2 Related Work

From time to time, various researchers have successfully applied several machine learning models for long and short-term prediction of air quality (Cabaneros et al. 2019,2017; Lightstone et al. 2017; Ibarra-Berastegi et al. 2008)[1][4][3]. For better prediction results, some research suggested using a vast dataset. Two-hybrid models had been proposed by Zhu et al., which were EMD-intrinsic mode functions (IMF) based hybrid model and empirical mode decomposition (EMD)-SVR based hybrid model. (Zhu et al. 2017)[5]. Jiaxuan Zhang et.al [7]used CNN-LSTM multi-model forecast Air Quality Index in Beijing, CNN's efficient feature extraction function was used to extract data features, improved the air quality prediction accuracy(Jiaxuan Zhang et al.2022).Sarun Duangsuwan used BPNN–CNN model predicted the results of air pollution parameters and generated 3D AQI mapping locations assess air pollution monitoring at 98% accuracy for data assessment in an open burning scenario (Sarun Duangsuwan et al. 2022)[2]. In our study, we propose a CNN network-based model to predict the relationship between air pollutants and AQI considering the influence of the geographical location of the observation sites.

3 Data

Our datasets are public data sets from EPA (United States Environmental Protection Agency) pre-Generated Data Files[6]. EPA programs use data standards to provide consistently defined and formatted data elements and sets of data values. These standards improve public access to meaningful environmental data. We used it to obtain data related to air pollution ($PM_{2.5}$, PM_{10}, SO_2, CO, and O_3) in Pittsburgh and sensors all across United States for 7 years (2015 to 2021), which also contained some labels like units of measure, observation count, observation percent, arithmetic mean, first maximum value, etc. For our prediction, we generally use the daily "Arithmetic Mean" values for all the pollutants. From the original data set (which is composed of thousands of different sensors for 1 year per CSV file, with some null values or no record for certain days), we extracted AQI and the daily average pollutants as labels through data engineering and used them in the machine learning process. Data engineering as part of the methodology of this project is described in detail subsequently.

4 Methods

The flowchart of the proposed model is shown in Fig. 1:

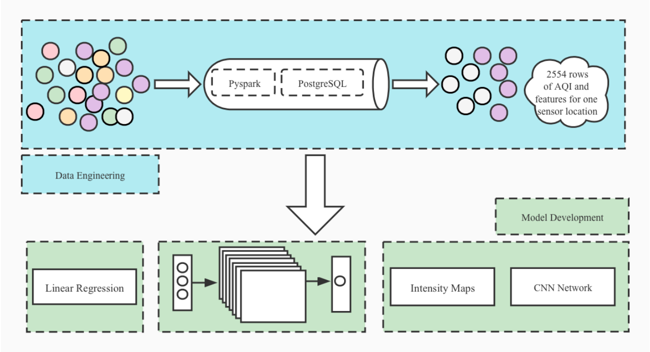
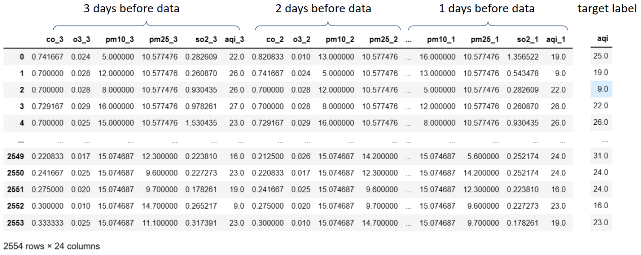


Figure 1: flowchart of the proposed model (Pat 1: Linear Regression, Part 2: DNN model, Part 3: CNN model)

4.1 Data Pre-processing and Data Pipeline

The presence of noise, outliers, and nulls in our dataset can affect the prediction process in a negative way, resulting in less accurate results. So firstly, we performed data pre-processing, using both PySpark and PostgreSQL tools, and put all the pollutants CSV data files obtained from the EPA Pre-Generated database (daily), and sorted all the monitor records by locations (Latitude, Longitude), POC (sensor number in the site), and filtered columns as ”Sample Duration: 1 HOUR”,”Event Type: None”. For different pollutants, the number of sites are different. Next we drop the sites that has less than 2350 row records (which is near 90% not-null recording out of 7 years, 2557 days). Taking CO as example, we got 358 sites after location filter, and 276 sites left after NA-drop. And then reindexed the sites corespond to 7 years constantly, made up for no-record days with the average value in the site. After data pre-process, Each site has 2557 rows, indexed with date from 2015-01-01 to 2021-12-31.



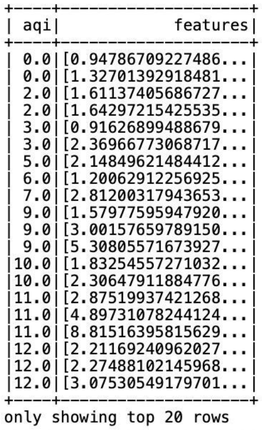
	3 days before data						2 days before data					1 days before data				target label
	co_3	**o3_3**	**pm10_3**	**pm25_3**	**so2_3**	**aqi_3**	**co_2**	**o3_2**	**pm10_2**	**pm25_2**	**...**	**pm10_1**	**pm25_1**	**so2_1**	**aqi_1**	**aqi**
0	0.741667	0.024	5.000000	10.577476	0.282609	22.0	0.820833	0.010	13.000000	10.577476	...	16.000000	10.577476	1.356522	19.0	25.0
1	0.700000	0.028	12.000000	10.577476	0.260870	26.0	0.741667	0.024	5.000000	10.577476	...	13.000000	10.577476	0.543478	9.0	19.0
2	0.700000	0.028	8.000000	10.577476	0.930435	26.0	0.700000	0.028	12.000000	10.577476	...	5.000000	10.577476	0.282609	22.0	9.0
3	0.729167	0.029	16.000000	10.577476	0.978261	27.0	0.700000	0.028	8.000000	10.577476	...	12.000000	10.577476	0.260870	26.0	22.0
4	0.700000	0.025	15.000000	10.577476	1.530435	23.0	0.729167	0.029	16.000000	10.577476	...	8.000000	10.577476	0.930435	26.0	26.0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
2549	0.220833	0.017	15.074687	12.300000	0.223810	16.0	0.212500	0.026	15.074687	14.200000	...	15.074687	5.600000	0.252174	24.0	31.0
2550	0.241667	0.025	15.074687	9.600000	0.227273	23.0	0.220833	0.017	15.074687	12.300000	...	15.074687	14.200000	0.252174	24.0	24.0
2551	0.275000	0.020	15.074687	9.700000	0.178261	19.0	0.241667	0.025	15.074687	9.600000	...	15.074687	12.300000	0.223810	16.0	24.0
2552	0.300000	0.010	15.074687	14.700000	0.265217	9.0	0.275000	0.020	15.074687	9.700000	...	15.074687	9.600000	0.227273	23.0	16.0
2553	0.333333	0.025	15.074687	11.100000	0.317391	23.0	0.300000	0.010	15.074687	14.700000	...	15.074687	9.700000	0.178261	19.0	23.0

2554 rows × 24 columns

Table 1: (Pittsburgh) Data structure for prediction with 3, 2 days and 1 day ago values, correspond to target AQI (today), after data pre-processing

Second, the pre-processed data for all pollutants were put into our data pipeline. For Linear Regression model and Deep Neural Network model, the 3,2 days and 1 days ago data of CO, O_3, $PM_{2.5}$ and other substances in Pittsburgh air that may affect air quality were loaded, and merged with the target AQI of Pittsburgh (we can call the target as ”Today AQI”). A total of 2554 rows and 24 columns of raw data were obtained (Table 1). The pyspark pipeline is composed with 4 data processing stages: FeatureTypeCaster(Change data feature to Double type), VectorAssembler(take all input columns as one vector), StandardScaler(Standardize the features), ColumnDropper(Drop all columns not in use).

Third, for CNN model, the pre-processed data for Ozone were put into our data pipeline, we would want to keep the geography information and the pollutant interacting information between Pittsburgh and Other cities, a 40x50x3 map is created for everyday data and maps for every three days are stacked to form an ”image”. The 40x50 map is bounded by the maximum and minimum of the coordinates (and the 3 depth dimension is the 3, 2 days and 1 days ago data) with the input of daily ozone concentrations. Just the same with step 2, the data is pipeline processed and the target is still ”Today” AQI of Pittsburgh. (Figure 4)



```
+----+--------------------+
| aqi|            features|
+----+--------------------+
| 0.0|[0.94786709227486...|
| 0.0|[1.32701392918481...|
| 2.0|[1.61137405686727...|
| 2.0|[1.64297215425535...|
| 3.0|[0.91626899488679...|
| 3.0|[2.36966773068717...|
| 5.0|[2.14849621484412...|
| 6.0|[1.20062912256925...|
| 7.0|[2.81200317943653...|
| 9.0|[1.57977595947920...|
| 9.0|[3.00157659789150...|
| 9.0|[5.30805571673927...|
|10.0|[1.83254557271032...|
|10.0|[2.30647911884776...|
|11.0|[2.87519937421268...|
|11.0|[4.89731078244124...|
|11.0|[8.81516395815629...|
|12.0|[2.21169240962027...|
|12.0|[2.27488102145968...|
|12.0|[3.07530549179701...|
+----+--------------------+
only showing top 20 rows
```

Table 2: Structure of data, after data pipeline.fit

4.2 Linear Regression (Pyspark ML)

Linear regression is a commonly used method in machine learning that is used to model the relationship between a dependent variable and one or more independent variables. In a linear regression model, the dependent variable is predicted by a linear combination of the independent variables.The mean squared error (MSE) is a common metric used to evaluate the model's performance. It is calculated as the average of the squared differences between the predicted values and the true values, and is a measure of the overall error of the model, which is able to predict the true values of the dependent variable. A lower MSE value indicates that the model is making more accurate predictions, while a higher MSE value indicates that the model is less accurate. So in our project, we use MSE in combination with other metrics together, these metrics can provide a comprehensive evaluation of a linear regression model's performance.

We used a linear regression model ,regressed the data linearly with hyperparameter adjustment,and choose the addGrid(lr.regParam) is 0.001, 0.01, 0.1, 1; choose the addGrid(lr.maxIter) is 100, 200, 500,1000 when parallelism=8, numFolds=5, and obtained the value of Train MSE and Test MSE. From the result we choose the best regParam as 0.01 and betst maxIter as 1000.

4.3 Deep Neural Network (Keras)

Keras is an open-source deep learning library that is written in Python and is capable of running on top of other popular deep learning libraries, such as TensorFlow, Theano, and CNTK. In Keras, a neural network is represented as a sequence of layers, which are stacked on top of each other to form a deep learning model. Each layer is a modular, self-contained building block that can be combined with other layers to create a deep learning architecture. Keras makes it easy to develop and train neural networks using a variety of different architectures.

In this part, we use TensorFlow and Keras to build our neural network to process the data with linear layers. The learning rate, epochs, loss, and val_loss of this data set are calculated and obtained learning rate, epochs, loss, and val_loss of the best model.Epoch in our model refers to the one entire passing of training air data through the algorithm. It's a hyperparameter defined as the total number of iterations of all the training data in one cycle for training the machine learning model. The learning rate is also a hyper-parameter that controls how much we are adjusting the weights of our network with respect to the loss gradient. Loss is a number of the penalty for a bad prediction indicating how bad the model's prediction was on a single example. If the model's prediction is perfect, the loss is zero; otherwise, the loss is greater. val_loss is the value of the cost function for your cross-validation data and loss is the value of the cost function for your training data. So in our model, we later determine the results of model training by the hyperparameters.

Here we just use a simple Deep Neural Network. (model = keras.Sequential(), keras.layers.Dense(wid,activation='relu'), Optimizer= Adam(0.01), epoch=100). We chooses MSE fr the loss function and evaluator parameter. And hyperparameter tunned the model for Layers depth in [3,5,7,10] and width in [15,18,20,25].

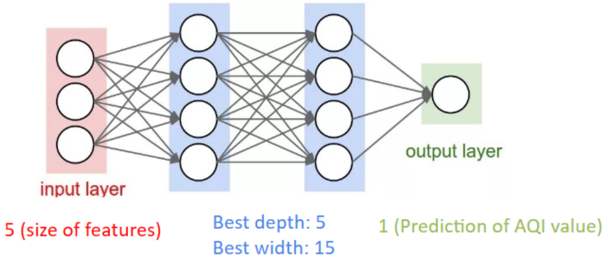


Figure 2: Structure of neural networks

4.4 Convolutional Neural Network

We figured that it is not enough to only do analysis based on the pollutants' concentration in Pittsburgh itself, but should lay more emphasis on the location of the monitors and the meteorological data, and thus we are trying to create intensity maps across the United States and train them using CNN to predict the AQI of Pittsburgh.

Convolutional Neural Network (CNN) is a type of deep learning neural network that is designed to automatically and adaptively learn spatial hierarchies of features from input data by using multiple layers of interconnected nodes. CNNs are composed of multiple layers, including an input layer, hidden layers, and an output layer. The hidden layers consist of multiple convolutional layers and pooling layers, which extract features from the input data and reduce its dimensionality; these layers combine the extracted features to make predictions. In this part, we use the CNN model to do predictions. Data-driven models are generally dependent on the size of the dataset. CNN models, like other empirical models, can be applied to datasets of arbitrary size, but the dataset used for training should be large enough to cover all known possible problems in the problem domain.

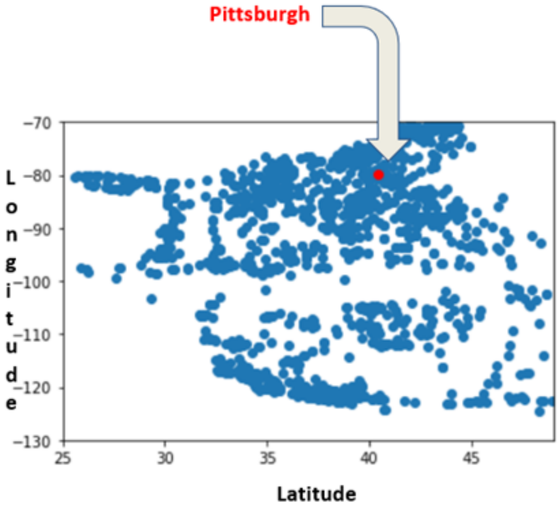


Figure 3: Geographic location of each monitoring station.

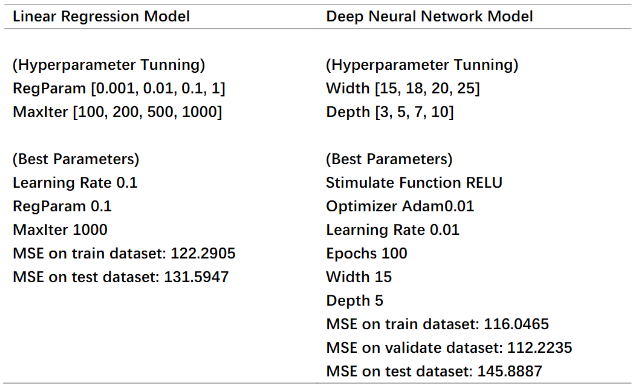
The datasets we are using are sensor-based air quality data and geographical location data. Based on the empirical research, the best results are obtained by partitioning the dataset into 7-80% for training and 20-30% for testing. That's why the dataset in this work is bisected as 80% for training and 20% for testing purposes. Our Linear Regression model and Convolutional Neural Network (CNN) model are trained with the training dataset and tested on the testing dataset to evaluate the results for analyzing the performance of the models. For DNN model, the train, validate and test data set is randomly splited as 80%, 10% and 10%. Further subsections discuss the implementation of the models and the analysis of results.

4.5 Linear Regression Model & DNN Model

We tested both linear regression models and a neural network model for the air quality index estimation. After several trials, we found that this two simple models is not good at predict the air quality index. The MSE is far from ideally closing to 0.

The pre-processed air quality data are provided as input to the models. The training process uses labeled data and allows the machine to learn the relationship between features and labels by comparing the error between the predicted value and the actual label and using a gradient descent algorithm to reduce the loss, tuning the parameters in continuous iterations until the algorithm finds the model parameters with the lowest possible loss or the overall loss stops changing and the optimal model is determined; while the prediction process applies the trained model to the unlabeled data, i.e., mapping the data to the predicted labels. The gradient descent algorithm uses a gradient multiplied by a learning rate to determine the location of the next point in the loss function image. The size of the set learning rate is related to the degree of gradient flatness. Batch size is the total number of data used to compute the gradient in a single iteration, and selecting small batches is usually more effective than whole batches when performing gradient descent on large datasets.

To facilitate multiple hyperparameter tuning of the model, we created functions for building and training the model, and determined hyperparameters such as learning rate and epochs after several tests. The final loss curve shows a steady decline with a slope close to zero, indicating that the model converges. At last, the performance assessment of the Linear Regression model is presented using the MSE method. The obtained MSE values are listed below.



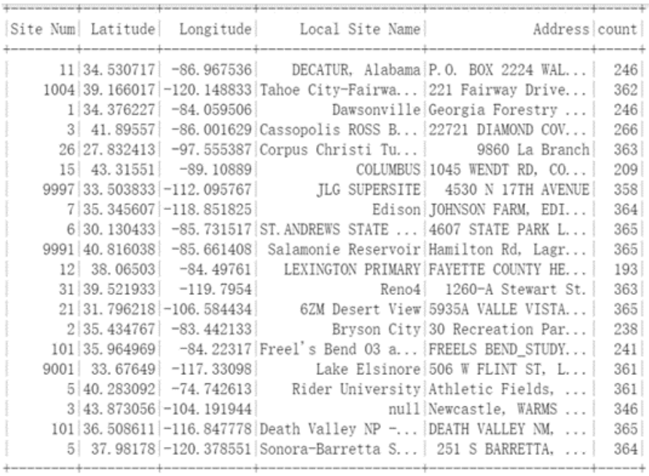
Linear Regression Model	Deep Neural Network Model
(Hyperparameter Tunning)	(Hyperparameter Tunning)
RegParam [0.001, 0.01, 0.1, 1]	Width [15, 18, 20, 25]
MaxIter [100, 200, 500, 1000]	Depth [3, 5, 7, 10]
(Best Parameters)	(Best Parameters)
Learning Rate 0.1	Stimulate Function RELU
RegParam 0.1	Optimizer Adam0.01
MaxIter 1000	Learning Rate 0.01
MSE on train dataset: 122.2905	Epochs 100
MSE on test dataset: 131.5947	Width 15
	Depth 5
	MSE on train dataset: 116.0465
	MSE on validate dataset: 112.2235
	MSE on test dataset: 145.8887

Table 3: MSE values of four prediction models.

Note that the AQI on the next day is predicted using the previous 2-day information.

4.6 Convolutional Neural Network Model

As the normal model (Linear Regression and DNN are not performed well in the prediction task, we keep our eyes on the CNN, since the CNN model has more divisions and may contain the interregional interaction between Pittsbrugh and other cities, which could influence the model performance. To investigate whether interregional pollutant transportation has a relatively large impact on air quality in Pittsburgh, we use CNN to perform complex mathematical calculations on the input time series, identifying useful information and feature extraction. As explained in 4.Methods, the datasets of air pollutant concentrations for Pittsburgh and the surrounding area are available from the EPA website, which includes the geographic location (latitude and longitude) of each monitoring station.



```
+--------+---------+-----------+-------------------+-------------------+-----+
|Site Num| Latitude| Longitude |   Local Site Name |            Address|count|
+--------+---------+-----------+-------------------+-------------------+-----+
|      11|34.530717| -86.967536|   DECATUR, Alabama|P.O. BOX 2224 WAL...|  246|
|    1004|39.166017|-120.148833|Tahoe City-Fairwa...|221 Fairway Drive...|  362|
|       1|34.376227| -84.059506|        Dawsonville|Georgia Forestry ...|  246|
|       3| 41.89557| -86.001629|Cassopolis ROSS B...|22721 DIAMOND COV...|  266|
|      26|27.832413| -97.555387|Corpus Christi Tu...|      9860 La Branch|  363|
|      15| 43.31551|  -89.10889|           COLUMBUS|1045 WENDT RD, CO...|  209|
|    9997|33.503833|-112.095767|     JLG SUPERSITE| 4530 N 17TH AVENUE|  358|
|       7|35.345607|-118.851825|             Edison|JOHNSON FARM, EDI...|  364|
|       6|30.130433| -85.731517|ST.ANDREWS STATE ...|4607 STATE PARK L...|  365|
|    9991|40.816038| -85.661408| Salamonie Reservoir|Hamilton Rd, Lagr...|  365|
|      12| 38.06503|  -84.49761|  LEXINGTON PRIMARY|FAYETTE COUNTY HE...|  193|
|      31|39.521933|  -119.7954|              Reno4|  1260-A Stewart St.|  363|
|      21|31.796218|-106.584434|    6ZM Desert View|5935A VALLE VISTA...|  365|
|       2|35.434767| -83.442133|        Bryson City|30 Recreation Par...|  238|
|     101|35.964969|  -84.22317|Freel's Bend 03 a...|FREELS BEND_STUDY...|  241|
|    9001| 33.67649| -117.33098|      Lake Elsinore|506 W FLINT ST, L...|  361|
|       5|40.283092| -74.742613|   Rider University|Athletic Fields, ...|  361|
|       3|43.873056|-104.191944|               null|Newcastle, WARMS ...|  346|
|     101|36.508611|-116.847778|Death Valley NP -...|DEATH VALLEY NM, ...|  365|
|       5| 37.98178|-120.378551|Sonora-Barretta S...| 251 S BARRETTA, ...|  364|
+--------+---------+-----------+-------------------+-------------------+-----+
```

Table 4: Geographic location (latitude and longitude) of each monitoring station.

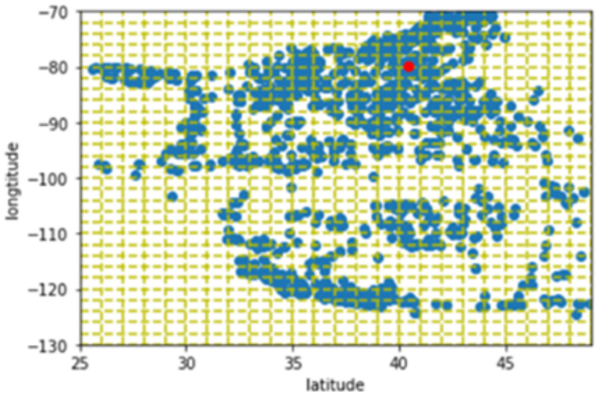


Figure 4: The 2-d map with adjacent sites falling into the same cell.

Since CNN has internal structures that are designed to operate upon three-dimensional image data, a 40x50 map is created for everyday data and maps for every three days are stacked to form an ”image”. The map is bounded by the maximum and minimum of the coordinates, with the input of daily ozone concentrations. If multiple locations are assigned to the same point, their input would be averaged. The ozone concentrations in Pittsburgh are the target values (except that of the first three days). These intensity maps are generated using OpenCV and then provided as input to the CNN model. The CNN model is run using Adam optimizer with a learning rate of 1e-5, and the batch size is set to 1 for better convergence.

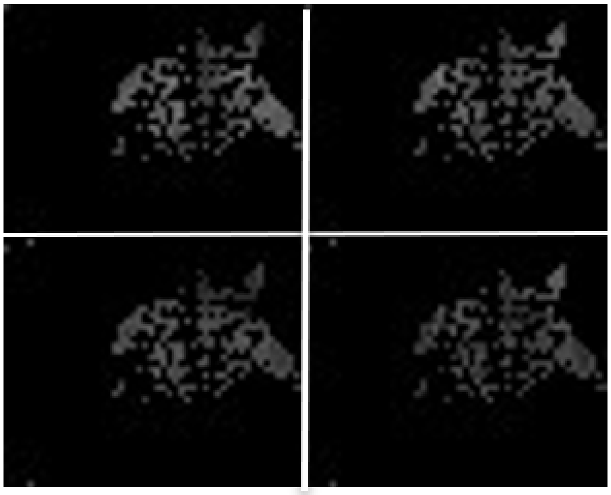


Figure 5: Intensity maps generated using OpenCV.

The losses of the training and testing process are calculated and shown below. As we can see, the train and test loss for CNN model is less than 0.1, and the plots of loss cross to iterations show that changes in pollutant concentrations in Pittsburgh are significantly influenced by air pollution in the surrounding area.

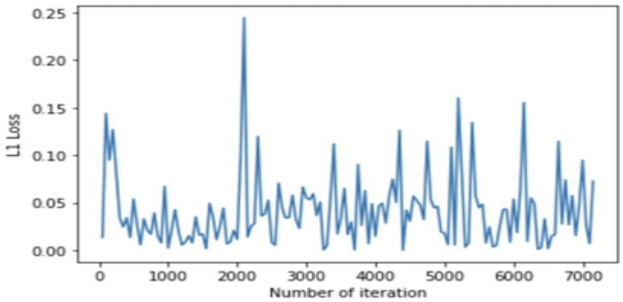


Figure 6: Train loss vs. number of iterations.

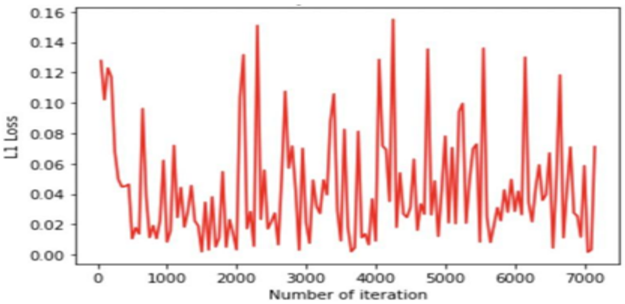


Figure 7: Test loss vs. number of iterations.

5 Conclusion

The results of the prediction evaluation show that linear regression and DNN are not a good model for predicting Pittsburgh's air quality, while our CNN model performs very well in the prediction, and the loss curve of our CNN model may indicate that interregional pollutant transportation can have a great impact on the air quality of Pittsburgh.

However, since the monitor/sensor sites are not exactly same for different Pollutants, our CNN model only use Ozone and it's AQI as input features, the conclusion that CNN model performs better for AQI prediction may change if all pollutants are take into consideration.

In the future, our work could be extended in the following aspects. First, we will consider more data related to air quality, e.g., various urban big data related to air quality (meteorology, traffic, factory air pollutant emissions, road network distribution, etc.) Second, other approaches, such as semi-supervised learning and Long Short Term Memory (LSTM) could be introduced to improve the accuracy and practicability of the model. Third, we could evaluate our method in more cities to make the results more convincing.

6 Labor division

The members of our team are Wenyan Zheng, Haineng Huang, JianFeng Su, and Kaixin Song. Our TA advisor is Linji Wang. Our tasks mainly include data processing, model development and evaluation. These were evenly distributed to each member.

Data processing and model development: all team members

Final report write-up and presentation slides: all team members

References

[1] Sheen Mclean Cabaneros, John Kaiser Calautit, and Ben Richard Hughes. "A review of artificial neural network models for ambient air pollution prediction". In: *Environmental Modelling & Software* 119 (2019), pp. 285–304.

[2] Sarun Duangsuwan et al. "3D AQI Mapping Data Assessment of Low-Altitude Drone Real-Time Air Pollution Monitoring". In: *Drones* 6.8 (2022), p. 191.

[3] Gabriel Ibarra-Berastegi et al. "From diagnosis to prognosis for forecasting air pollution using neural networks: Air pollution monitoring in Bilbao". In: *Environmental Modelling & Software* 23.5 (2008), pp. 622–637.

[4] Helgi I Ingólfsson et al. "Computational lipidomics of the neuronal plasma membrane". In: *Biophysical journal* 113.10 (2017), pp. 2271–2280.

[5] Phillip Isola et al. "Image-to-image translation with conditional adversarial networks". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 1125–1134.

[6] OAR US EPA. *Air Data: Air Quality Data Collected at Outdoor Monitors Across the US*. en. Collections and Lists. July 2014. URL: `https://www.epa.gov/outdoor-air-quality-data` (visited on 11/11/2022).

[7] Jiaxuan Zhang and Shunyong Li. "Air quality index forecast in Beijing based on CNN-LSTM multi-model". In: *Chemosphere* 308 (2022), p. 136180.