COMP 432 Project Report

Anomaly detection for PM2.5 concentration in Beijing

Li Sun 40017648 Feng Zhao 40021856

Abstract

Anomaly detection is the identification of abnormal items, events or observations which raise suspicions by differing significantly from the majority of the data. The main goal of this project is to explore anomaly detection performance with unsupervised machine learning using autoencoder in reviewing the documented concentration of particulate matter (PM) with a diameter of less than 2.5 micrometers (PM2.5) in Beijing between Jan 1st, 2010 to Dec 31st, 2014. Using threshold 3.0, anomaly data generated RSME outputs of approximately 1.1-1.2 in comparison to normal data RMSE values of approximately 0.6, which suggests that the trained autoencoder can be used to separate anomaly data from normal data, and therefore support the identification to identify abnormally high PM2.5 values in Beijing.

Introduction

Due to rapid urban growth, air pollution has been one of the most serious environmental problems in Beijing, the capital of China. Particulate matter (PM) is the major air pollutant in Beijing and is causing serious ongoing threats to human health. Based on historical data, the average concentration of PM with a diameter of less than 2.5 micrometers (PM2.5) in Beijing between Jan 1st, 2010 to Dec 31st, 2014 PM is approximately 100 $\mu g/m^3$[1] This is approximately three times higher than the WHO Level 1 Interim Target of 35 $\mu g/m^3$[1]. By identifying outliers in this dataset, an anomaly detection system can identify observations that do not conform to normal behaviours, which in turn may support the city to highlight specific dates or periods of days where anomaly had previously occurred. In turn, this information can be used to support the identification of potential contributors or factors of abnormally high PM2.5 concentrations, and therefore support potential solutions to reduce future PM2.5 peaks. The goal of this project is to explore how an autoencoder may behave when applied to anomaly detection. The dataset we have selected is a tabular time series with numerical modalities. It contains 12 features (Figure 1), including timestamp, year, month, day, hour, PM2.5 concentration, dew point (DEWP), temperature (TEMP), pressure (PRES), combined wind direction (cbwd), cumulated wind speed (lws)(m/s), cumulated hours of snow (ls) and cumulated hours of rain (lr). It is anticipated that each data row will be

evaluated against an abnormal detection threshold set at three standard deviations away from the PM2.5 mean value of the dataset.

Methodology & Experimental Results

This study utilizes unsupervised machine learning technique of applying an autoencoder to the data sample, which consists of hourly PM2.5 concentration measurements at the US Embassy in Beijing and the associated meteorological data from the Beijing Capital International Airport between Jan 1st, 2010 to Dec 31st, 2014. The data was preprocessed and tagged, and then randomly shuffled to split into 80% training data and 20% testing data. After preprocessing, data anomalies in the training dataset was removed since anomaly detection is not a component of autoencoder training. The non-abnormal values within the training data was then fed into the training algorithm as a model input to build an autoencoder which will show a low reconstruction error when the future input is normal, but high reconstruction error when the future input is abnormal. The remaining 20% testing data was then fed to the autoencoder to generate reconstruction error and validate anomaly detection accuracy.

During the initial data preprocessing, abnormal data was labelled by different rolling window sizes (12 hours, 24 hours, 48 hours, 144 hours), inspired by Jason Brownlee (How to Convert a Time Series to a Supervised Learning Problem)[2]. For example, for a rolling window of 144 hours (7 days), the mean PM2.5 value on any given 8th day was compared to the mean PM2.5 value of the previous 7 days, for all days within the dataset. The distribution of the differences in means were then analyzed to identify values above 3 standard deviations from the mean of means. The difference between any mean daily PM2.5 values when compared to the mean PM2.5 value of the previous 7 days that were equal or greater than 3 deviations away from the mean of means from a 144 hour rolling window was then labelled as an anomaly, and therefore abnormal, in the original dataset. Following this process, 4 preprocessed data sets emerged after labelling anomalies by different rolling window sizes. The 144 hour rolling window dataset was selected to proceed for autoencoder training as it had the greatest number of detected anomalies out of the 4 preprocessed data sets. Figure 2 shows the output after the 144 hour rolling window data preprocessing. Anomalies (anom) were tagged as 1.

Following the tagging of daily data using the 144 hours rolling window approach, anomalies were plotted across the time series dataset to visualize the location of the outliers (Figure 3).

To build confidence that the tagged data share certain particularities or characteristics in order to apply neural networks technicals to separate abnormal data from normal data,

a t-distributed stochastic neighbour embedding (t-SNE) graph was generated (Figure 4). The graph shows a high level of similarity between the tagged anomaly data.

A 5 hidden layer model was selected to balance computational resources required and increased performance of higher layer models. The need to determine the numbers of neutrals in each layer also led to a hyperparameter search. Three types of parameters were searched: outer layer neural numbers, middle layer neutral numbers, and code layer neutral numbers, resulting in a total of 120 models (Figure 5).

The models were evaluated using AUROC precision-recall curves. Although a function to calculate AUROC can be implemented, a rough visual check was practiced for this experiment (Figure 6). The full version of precision-recall curves can be shown when implementing the code.

The highest AUROC value is a 12-11-5-2-5-11-12 structure. It has a total of 440 trainable parameters with training losses (RMSE) of approximately 0.42-0.43 (Figure 7).

The anomaly data which did not participate in training are used as input to the best model to check its RMSE, which generated an output of approximately 1.1-1.2 during several trainings, whereas reconstructed normal data generated RMSE values of approximately 0.6.

Since an “expertise” threshold value is not available, a comparison of various threshold values was tested and 3.0 was chosen to make an acceptable classifier. Below show the results for the testing part in one run (Figure 8).

Conclusions

Using threshold 3.0, a reconstruction with precision 0.53 and recall 0.68 was generated in one of the runs. Since anomaly data generated outputs of approximately 1.1-1.2 in comparison to normal data RMSE values of approximately 0.6, this demonstrates that the trained autoencoder has difficulty in reconstructing the anomaly data, and thus this autoencoder can be used to separate anomaly data from normal data. Further measures can be used to improve the performance of our autoencoder such as modifying the structure of it, running more epoches or using a different loss function.

Group 38

References

[1] Wu, J., Li, J., Peng, J., Li, W., Xu, G. and Dong, C., 2014. Applying land use regression model to estimate spatial variation of PM2.5 in Beijing, China. Environmental Science and Pollution Research, 22(9), pp.7045-7061.
[2] Brownlee, J., 2020. How To Convert A Time Series To A Supervised Learning Problem In Python. [online] Machine Learning Mastery. Available at: <https://machinelearningmastery.com/convert-time-series-supervised-learning-problem-python/>.
[3] Kaggle.com. 2020. Anomaly Detection With Auto-Encoders. [online] Available at: <https://www.kaggle.com/robinteuwens/anomaly-detection-with-auto-encoders>.

Appendix

List of Figures

timestamp	year	month	day	hour	pm2.5	DEWP	TEMP	PRES	cbwd	Iws	Is	Ir
2010-01-18 00:00:00	2010	1	18	0	282.0	-12	-10.0	1028.0	cv	0.89	0	0
2010-01-18 01:00:00	2010	1	18	1	282.0	-13	-11.0	1028.0	NW	3.13	0	0
2010-01-18 02:00:00	2010	1	18	2	303.0	-11	-9.0	1028.0	NW	4.92	0	0
2010-01-18 03:00:00	2010	1	18	3	349.0	-13	-10.0	1028.0	NW	6.71	0	0
2010-01-18 04:00:00	2010	1	18	4	407.0	-13	-11.0	1028.0	NW	7.60	0	0
2010-01-18 05:00:00	2010	1	18	5	361.0	-13	-11.0	1027.0	NE	1.79	0	0
2010-01-18 06:00:00	2010	1	18	6	234.0	-14	-11.0	1028.0	NW	0.89	0	0

Figure 1. Selected Tabular Time Series Data Extract

timestamp	year	month	day	hour	pm2.5	DEWP	TEMP	PRES	cbwd	Iws	Is	Ir	anom	diff_pm_2.5
2010-01-18 00:00:00	2010	1	18	0	282.0	-12	-10.0	1028.0	1	0.89	0	0	0	182.361111
2010-01-18 01:00:00	2010	1	18	1	282.0	-13	-11.0	1028.0	2	3.13	0	0	0	180.548611
2010-01-18 02:00:00	2010	1	18	2	303.0	-11	-9.0	1028.0	2	4.92	0	0	0	199.701389
2010-01-18 03:00:00	2010	1	18	3	349.0	-13	-10.0	1028.0	2	6.71	0	0	0	243.458333
2010-01-18 04:00:00	2010	1	18	4	407.0	-13	-11.0	1028.0	2	7.60	0	0	1	298.736111
2010-01-18 05:00:00	2010	1	18	5	361.0	-13	-11.0	1027.0	3	1.79	0	0	0	250.291667
2010-01-18 06:00:00	2010	1	18	6	234.0	-14	-11.0	1028.0	2	0.89	0	0	0	121.743056

Figure 2. 144 Hour Rolling Window Data Preprocessing Outcome Extract

Group 38

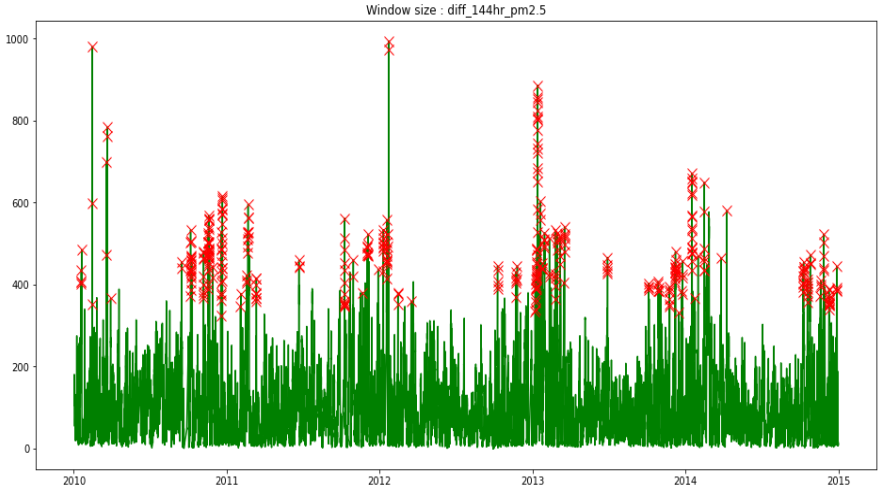

Figure 3. Outlier Visualization Across Time Series Dataset

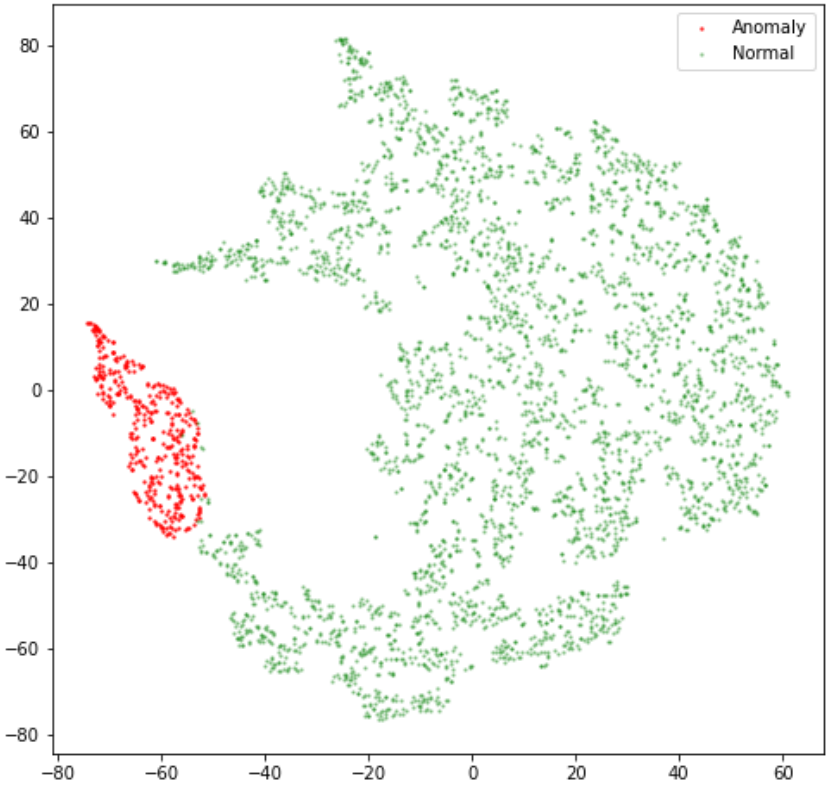

Figure 4. t-SNE Graph Showing Anomaly Data Similarities

Group 38

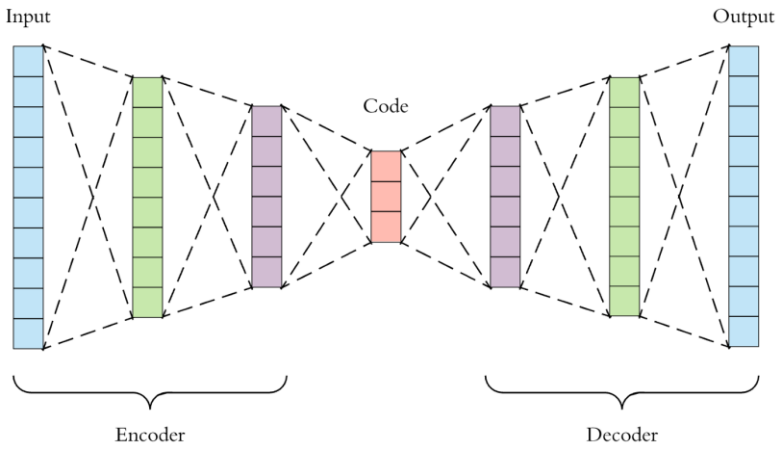

Figure 5. 5 Hidden Layer Model [3]

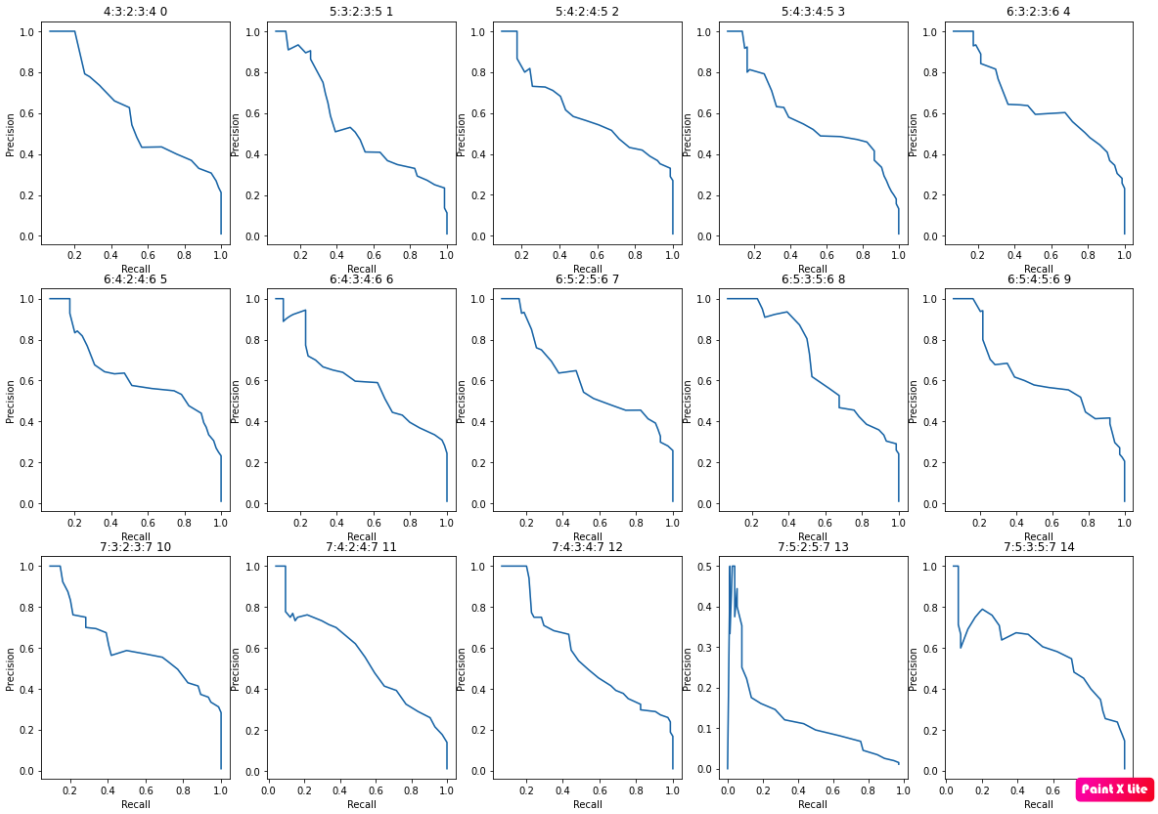

Figure 6. Sample of AUROC Precision-Recall Curves

Group 38

```
Model: "sequential_87"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
dense_522 (Dense)            (None, 11)                143
_________________________________________________________________
dense_523 (Dense)            (None, 5)                 60
_________________________________________________________________
dense_524 (Dense)            (None, 2)                 12
_________________________________________________________________
dense_525 (Dense)            (None, 5)                 15
_________________________________________________________________
dense_526 (Dense)            (None, 11)                66
_________________________________________________________________
dense_527 (Dense)            (None, 12)                144
=================================================================
Total params: 440
Trainable params: 440
Non-trainable params: 0
_________________________________________________________________
model 87 training loss = 0.4201982617378235
model 87 validation loss = 0.4340282380580902
```

Figure 7. RMSE Training Output Sample

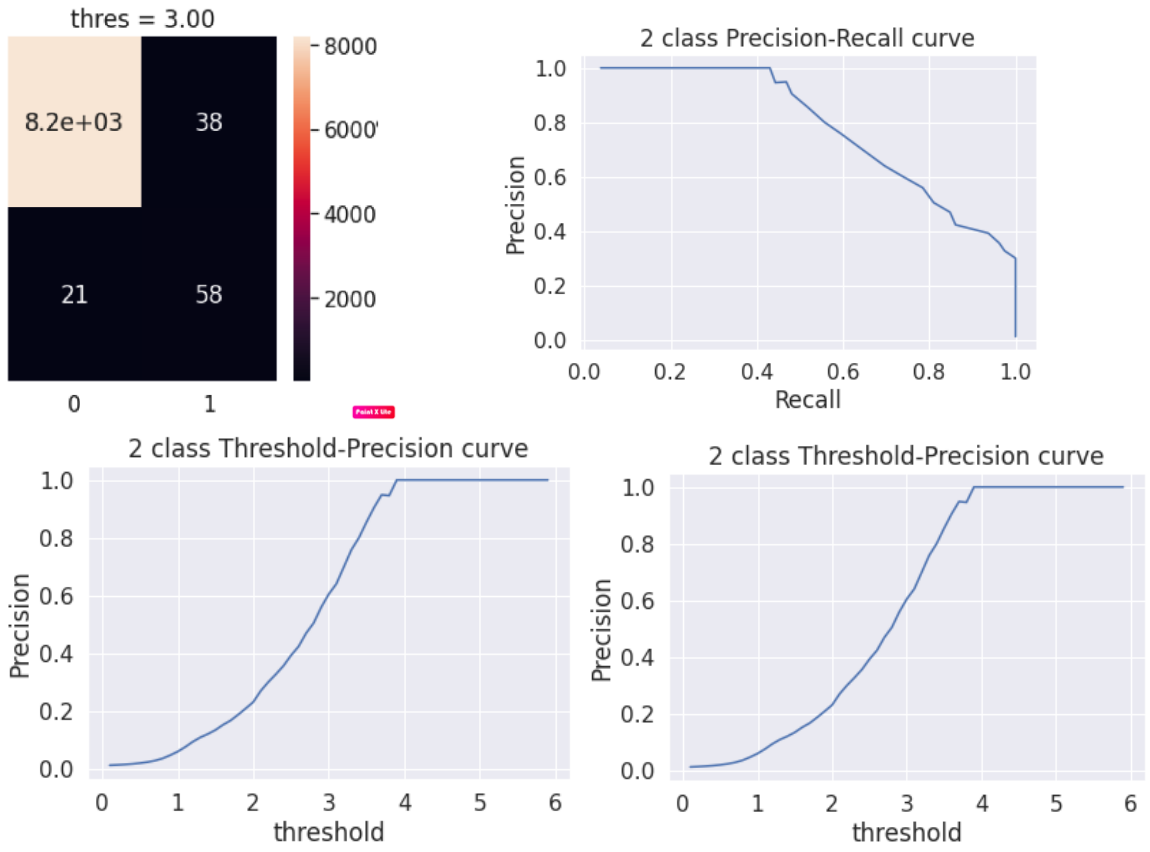

Figure 8. Confusion Matrix and Precision Recall Curve