Pollution Forecast Using LSTM and CNN-LSTM

Quinn Leydon
KGCOE
Rochester Institute of Technology
Rochester, NY.
qcl4604@rit.edu

***Abstract*— A machine learning model to predict hazardous pollution levels in Beijing China was investigated. Concentration of PM2.5 greater than 250.5 µg/m^3 is considered hazardous and occurs many times per year in China. The models investigated where LSTM and CNN-LSTM. The LSTM regression as used in prior research was found to under predict peak pollution levels and have an offset. The offset is the model adapting to the peak pollution when it was measured, resulting in a three-hour delay in predictions. Despite the acceptably low mean square error, the hour-to-hour prediction of pollution level was inadequate. The CNN-LSTM was less likely to underpredict peak pollution levels than the LSTM but was still not adequate. To align the training process with the goal of an early warning system, the model was converted into a binary classification model for hazardous levels. The conversion to classification removed the offset in the predicted values and had better accuracy. This paper advises using classification rather than regression for rare event prediction in timeseries data.**

Keywords—LSTM, CNN_LSTM, Pollution, PM2.5

I. Introduction

Air pollution is a substantial health concern in large urban areas. Air pollution can increase dangerous diseases and according to the World Health Organization, causes millions of deaths [1]. One of the most important pollutants is the PM2.5 concentration, which is particles in the air under 2.5 micrometers. PM2.5 is also a major contributing factor to haze. The United States EPA considers any prolonged exposure to concentration of PM2.5 greater than 250.5 µg/m^3 to be hazardous [2]. The severity of PM2.5 concentration is shown in Fig. 1.

AQI Category	Index Values	Previous Breakpoints (1999 AQI) (µg/m^3, 24-hour average)	Revised Breakpoints (µg/m^3, 24-hour average)
Good	0 - 50	0.0 - 15.0	0.0 – 12.0
Moderate	51 - 100	>15.0 - 40	12.1 – 35.4
Unhealthy for Sensitive Groups	101 – 150	>40 – 65	35.5 – 55.4
Unhealthy	151 – 200	> 65 – 150	55.5 – 150.4
Very Unhealthy	201 – 300	> 150 – 250	150.5 – 250.4
Hazardous	301 – 400	> 250 – 350	250.5 – 350.4
	401 – 500	> 350 – 500	350.5 – 500

Fig. 1. PM2.5 Hazard Levels [2]

There is a large quantity of research to forecast future pollution levels. [1] investigates future PM2.5 concentration using environmental conditions with a regression based Convolutional Neural Network (CNN), Long Short Term Memory (LSTM), and a combination of the two (CNN-LSTM). It found the CNN-LSTM was most effective for predicting future pollution levels. [3] Uses an LSTM regression to predict future pollution levels using measurements of known pollutants.

The goal is to predict future pollution levels to allow individuals to plan their transportation accordingly, to miss peak hours. By predicting high air pollution levels three hours in advance, individuals could avoid peak hours. The ability to predict peak hours was through both regression and classification between hazardous and nonhazardous levels.

II. Background

A. Dataset

The Beijing PM2.5 dataset was developed by the US embassy in Beijing to study the impact of environmental conditions on pollutant levels. The dataset measured hourly pollution, dew, temperature, pressure, wind direction, wind speed, snow, and rain over a 5-year period [4].

B. LSTM

LSTM is a sub-type of recursive neural network (RNN). These are neural networks which have memory cells. The LSTM differs from traditional RNNs by having a long-term memory function [1]. This allows the algorithm to have better results with timeseries data than traditional neural networks. The key component in the hidden layer of an LSTM consists of an input gate, a forget gate, and an output gate. The network is shown in Fig. 2 [5].

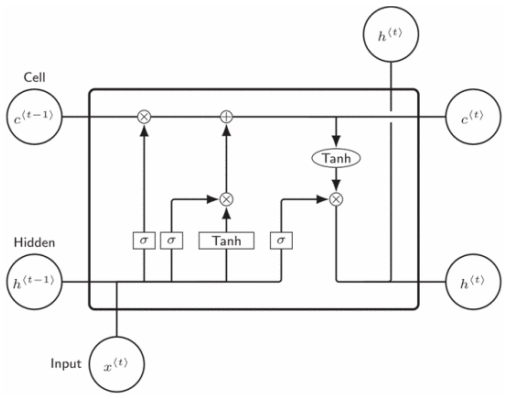

Fig. 2. LSTM Cell [5]

The input gate selects data to be stored for the next state. The forget gate selects data that will not be stored. This is important to limit the quantity of data required. The σ is the sigmoid function:

$$\frac{1}{1+e^{-x}} \quad [1]$$

Tanh function has a range from -1 to 1 and has a steeper slope than the sigmoid. The function is

$$\frac{e^{x}-e^{-x}}{e^{x}+e^{-x}} \quad [2]$$

C. CNN-LSTM

A CNN-LSTM is considered an enhanced version of an LSTM, where the LSTM takes its inputs from a 1d convolutional neural network (CNN) [6]. CNN and other deep learning networks struggle when there is an unbalanced data set [7]. An unbalanced data set occurs in classification when there is significantly less of one data class than others.

A 1D CNN applies many filters to multivariate timeseries data. These filters each result in a 1D output. The values within the filters are trained to enhance patterns in the data. By passing the enriched data to the LSTM, better results should ensue. Fig. 3 shows the 1D convolution [1].

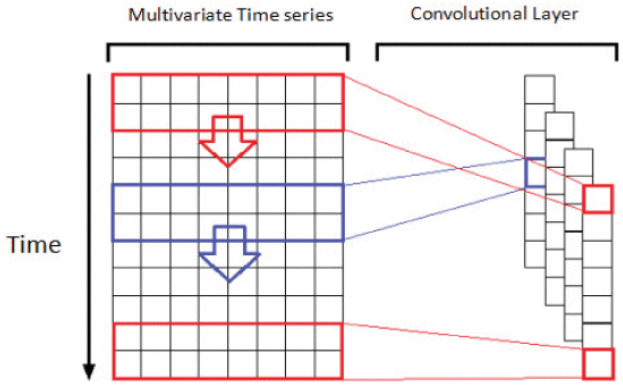

Fig. 3. 1D Convolutional Neural Network on Time Series Data [1].

III. Proposed Method

The proposed solution was an LSTM and CNN-LSTM evaluated for both regression and classification. First, the dataset was cleaned for missing data. The missing data was set to zero to allow for computation but may have decreased the accuracy of the model when compared to removing the timestep. The data was then standardized because the different features use different units. After standardization, the units should not have impact the weights for any of the features [8]. The dataset was standardized by using (3), where μ was the mean for the feature and σ was the standard deviation for the feature:

$$\frac{x-\mu}{\sigma} \quad [3]$$

The data was split for train, validation, test by (60,20,20). The was reshaped from [samples, features] to [samples, timesteps, features].

The model architecture was built using TensorFlow sequential layers. The first layer was an LSTM. The LSTM is followed by a dense layer and a dense layer of size one as the output. The CNN-LSTM begins with a 1D CNN.

To use classification rather than regression, the target values had to be filtered so that any value above the 250.5 threshold was set to 1 and any below was set to 0. The final output was be set to a sigmoid activation, which pushed the values to 0 or 1. A filter was then be applied to adjust any value between 0 and 1.

IV. Results

A. Dataset

The first 24 hours of the dataset did not have pollution values, so they were removed. The wind direction was removed. The features before preprocessing are shown in Fig. 4. There were 8 features and 43800 samples, resulting in a train length of 26,280 samples and a validation and test set of size 8760 samples. The wind direction was converted to radians. The target values were the pollution value shifted by 3 hours, reducing the possible amount of data by three. Filtering for classification results in an unbalanced data set with 3,053 hazardous hours out of 43,800 total hours, or 7% hazardous.

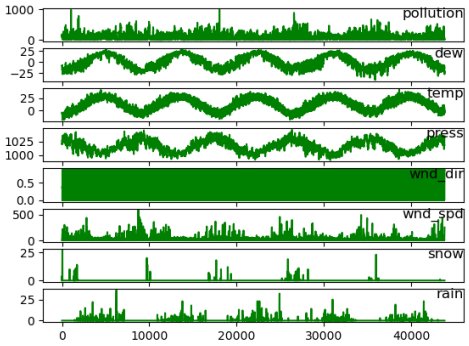

Fig. 4. Data Distribution

B. Hyper Parameters

The LSTM was identical to the CNN-LSTM without the LSTM Layer. The CNN used 64 filters, kernel size 2, with an input of 4 timesteps and 1 feature. The LSTM had 10 units and a ReLU activation function. This flowed into a 10-node dense layer and then a one node output layer. This was ReLU for regression and sigmoid for classification. The loss was mean absolute error, and the optimizer is Adam.

C. Algorithm

The training loss curve is shown in Fig. 5. The test quickly found a local minimum, then stays there.

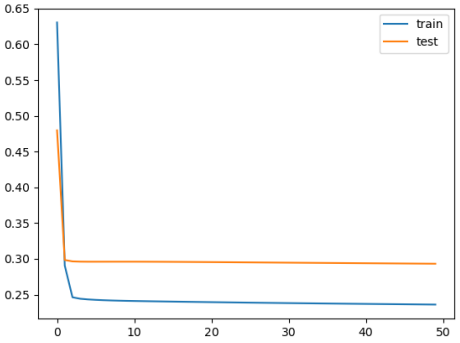

Fig. 5. Training for LSTM

The root square mean error was 42.489 for a 3-hour offset. The predicted pollution vs actual pollution shown in Fig. 6.

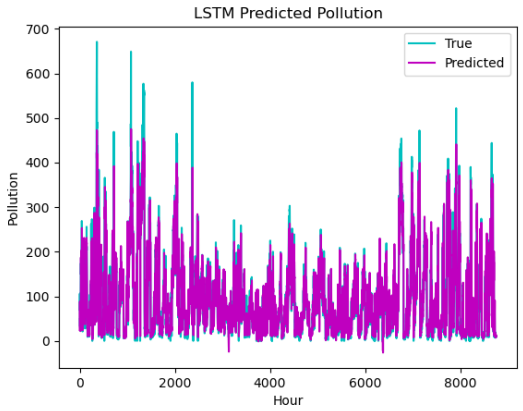

Fig. 6. Actual vs Predicted Pollution Levels LSTM.

The LSTM responded to peaks rather than predicted them. It also predicted lower peak values of pollution than the actual value. This was not ideal for predicting peak pollution. Fig. 7 shows a zoomed in section showing this miscalculation

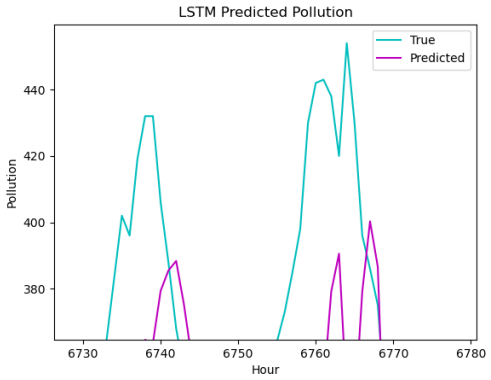

Fig. 7. Actual vs Predicted Pollution Levels with Zoom LSTM.

The CNN-LSTM had a RSME of 43.201. The actual pollution vs predicted pollution is shown in Fig. 8.

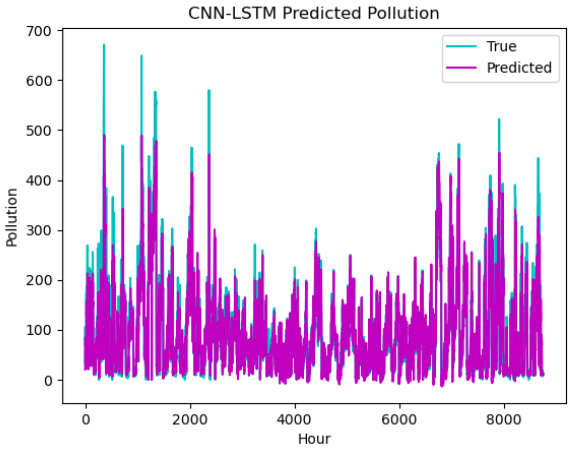

Fig. 8. Actual vs Predicted Pollution Levels CNN-LSTM.

The CNN-LSTM had higher peaks than the LSTM, but retained the issue of offset. Fig. 9 shows the CNN-LSTM under the same window as the LSTM.

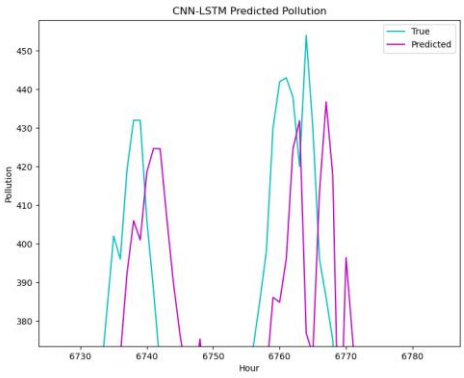

Fig. 9. Actual vs Predicted Pollution Levels with Zoom CNN-LSTM.

The Classification models did not retain the offset issue and were better at predicting high values of PM2.5 than the regression models. The actual vs predicted values of the LSTM are shown in Fig. 10.

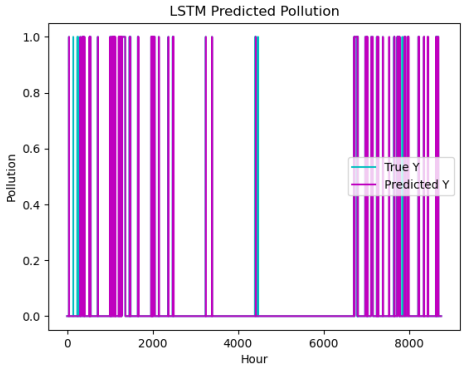

Fig. 10. Actual vs Predicted Pollution Levels LSTM Classification.

The LSTM had an accuracy of 0.9688, precision of 0.8062, and recall of 0.7784. The confusion matrix is shown in Fig. 11.

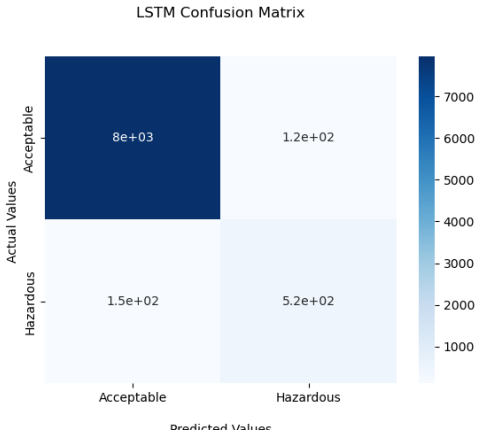

Fig. 11. Actual vs Predicted Pollution Levels LSTM.

The actual vs predicted values of the CNN-LSTM are shown in Fig. 12.

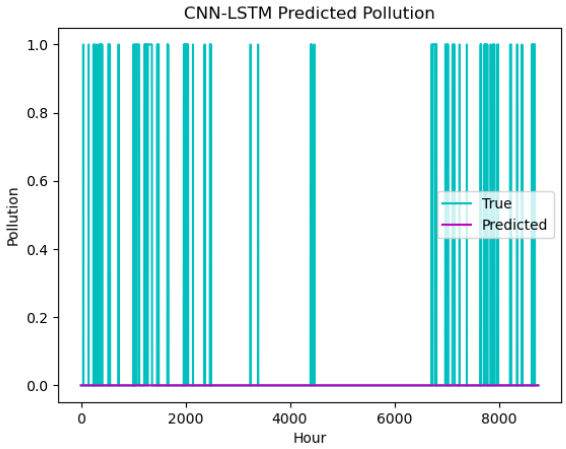

Fig. 12. Actual vs Predicted Pollution Levels CNN-LSTM.

The CNN-LSTM had an accuracy of 0.9237, precision of 0.0000, and recall of 0.0000. The confusion matrix is shown in Fig. 13.

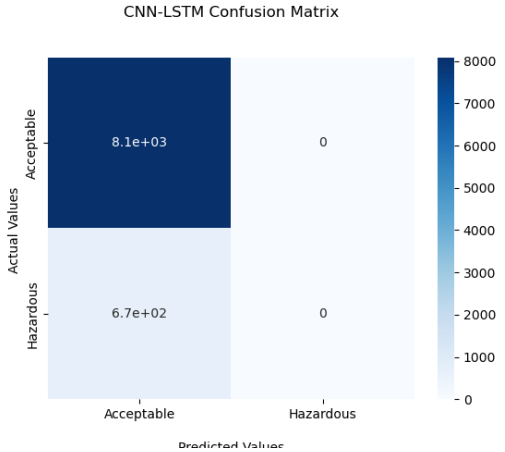

Fig. 13. Actual vs Predicted Pollution Levels LSTM.

V. Conclusion

Pollution forecasting using an LSTM and CNN-LSTM deep learning model was investigated. The model predicted the future pollution levels using current environmental data. The models were implemented using regression and classification. It was found that classification was a better metric for predicting high pollution levels. It was also found that the CNN-LSTM was unsuccessful in classifying the unbalanced data set.

The regression-based LSTM did not predict future pollution spikes but adapted its prediction based on observed spikes, which made it an inadequate model to use to predict future hazardous pollution levels. The LSTM also had the tendency to underpredict the peak values. The CNN-LSMT was less likely to underpredict peak values, which resulted in a larger mean square error, but better for predicting peak pollution.

Converting the model from regression to classification fixed most issues with the model. The delay in predicted peak was removed, resulting in adequate predictions. The recall of 0.7784 shows that most hazardous hours were being predicted and the precision of 0.8062 shows that most hazardous predictions were true positives. For this reason, it is recommended that classification should be used for rare event detection, such as peak pollution levels, instead of regression.

Acknowledgment

References

[1] T. Li, M. Hua and X. Wu, "A Hybrid CNN-LSTM Model for Forecasting Particulate Matter (PM2.5)," in IEEE Access, vol. 8, pp. 26933-26940, 2020, doi: 10.1109/ACCESS.2020.2971348.

[2] "The National Ambient Air Quality Standards for Particle Pollution," EPA, 14-Dec-2012. [Online]. Available: https://19january2021snapshot.epa.gov/sites/static/files/2016-04/documents/2012_aqi_factsheet.pdf. [Accessed: 13-Apr-2022].

[3] F. Hamami and I. A. Dahlan, "Univariate Time Series Data Forecasting of Air Pollution using LSTM Neural Network," 2020 International Conference on Advancement in Data Science, E-learning and Information Systems (ICADEIS), 2020, pp. 1-5, doi: 10.1109/ICADEIS49811.2020.9277393.

[4] R. Roy, "Air Pollution Forecasting - LSTM multivariate," *Kaggle*, 20-Jan-2022. [Online]. Available: https://www.kaggle.com/datasets/rupakroy/lstm-datasets-multivariate-univariate. [Accessed: 25-Apr-2022].

[5] K. Moharm, M. Eltahan and E. Elsaadany, "Wind Speed Forecast using LSTM and Bi-LSTM Algorithms over Gabal El-Zayt Wind Farm," 2020 International Conference on Smart Grids and Energy Systems (SGES), 2020, pp. 922-927, doi: 10.1109/SGES51519.2020.00169.

[6] N. Daoud, M. Eltahan and A. Elhennawi, "Aerosol Optical Depth Forecast over Global Dust Belt Based on LSTM, CNN-LSTM, CONV-LSTM and FFT Algorithms," IEEE EUROCON 2021 - 19th International Conference on Smart Technologies, 2021, pp. 186-191, doi: 10.1109/EUROCON52738.2021.9535571.

[7] W. Gao, L. Chen and T. Shang, "Stream of Unbalanced Medical Big Data Using Convolutional Neural Network," in IEEE Access, vol. 8, pp. 81310-81319, 2020, doi: 10.1109/ACCESS.2020.2991202.

[8] S. Lakshmanan, "How, when, and why should you normalize / standardize / rescale your data?," Towards AI, 16-May-2020. [Online]. Available: https://towardsai.net/p/data-science/how-when-and-why-should-you-normalize-standardize-rescale-your-data-3f083def38ff. [Accessed: 12-Apr-2022].