Air Quality Forecasting Using Convolutional LSTM

Shuo Sun & Gaoxiang Liu

{sunshuo, gaoxiang}@stanford.edu

Introduction

Air pollution is one of the biggest concerns of people, especially among developing countries. Being able to forecast air quality can help people prepare beforehand, and thus reduce the impact of hazardous weather. Therefore, we propose using Convolutional LSTM for air quality forecasting. The goal is that given the weather and air quality data of the past 24 hours, predict the air quality in the future 48 hours.

Data & Preprocessing

We use 1 year hourly weather and air quality data provided by KDD competition. The weather data is a series of 21 x 31 matrices with 5 real number weather data covers Beijing. The air quality data is a series of 6 pollutant concentration measures from 35 air quality stations. Weather and air quality data are of different scales and not centered around 0. This makes it hard for the model to learn the bias terms and getting signals from data effectively. Therefore, we apply normalization based on series (Figure 1).

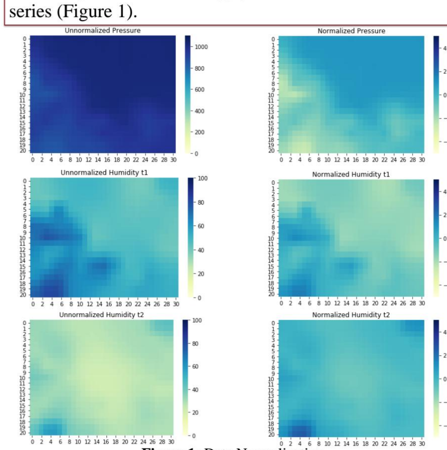


Figure 1: Data Normalization

Model Formulation

Traditional LSTM uses all values from the input and previous hidden state to compute the values for the next timestamp, which ignores spatial correlations. Convolutional LSTM uses convolution to calculate the gates and states. The mathematical representation and visualization are shown in Figure 2 and Figure 3.

The model takes in the normalized input tensor, and applies 3 layers of ConvLSTM to compute the state transition. Then it applies a `1x1` convolution on the ConvLSTM output to predict the next weather and air quality grid. Also, for each air quality station, the model takes the closest `3x3` tensor from the ConvLSTM output, convolves it into a vector, and then uses fully connected layers to predict the air quality measures for the station. (Figure 4)

$$i_t = \sigma(W_{xi} * X_t + W_{hi} * H_t + W_{ci} \circ C_{t-1} + b_i)$$
$$f_t = \sigma(W_{xf} * X_t + W_{hf} * H_t + W_{cf} \circ C_{t-1} + b_f)$$
$$o_t = \sigma(W_{xo} * X_t + W_{ho} * H_t + W_{co} \circ C_{t-1} + b_i)$$
$$C_t = f_t \circ C_{t-1} + i_t \circ \tanh(W_{xc} * X_t + W_{hc} * H_t + b_i)$$
$$H_t = o_t \circ \tanh(C_t)$$

Figure 2: Convolutional LSTM Formulae

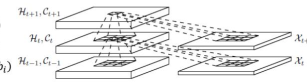


Figure 3: Convolutional LSTM Visualization [1]

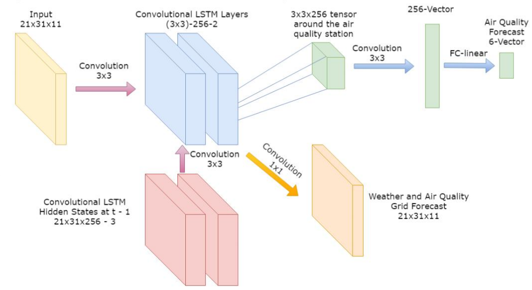
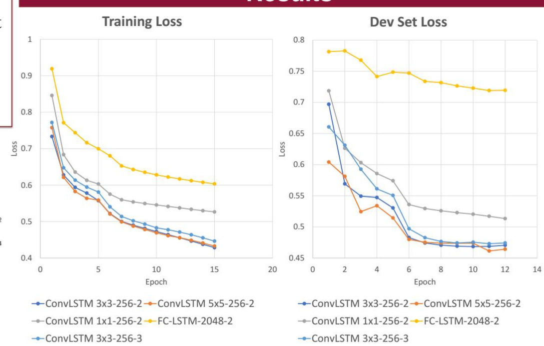


Figure 4: ConvLSTM Forecasting Model

Results



Model	Dev Loss	Test Loss	Test SMAPE
FC-LSTM-2048-2	0.71110	0.77009	0.45841
ConvLSTM 1x1-256-2	0.50785	0.60865	0.40018
ConvLSTM 3x3-256-2	**0.46798**	**0.56391**	**0.36646**
ConvLSTM 5x5-256-2	0.46131	0.56712	0.37865
ConvLSTM 3x3-256-3	0.47290	0.56988	0.38748

Conclusions & Discussions

- Comparing to FC-LSTM, by having much fewer parameters and utilizing sparsity of connection and parameter sharing, the ConvLSTM models learn much faster and perform much better.
- By comparing the results of `3x3` and `1x1` convolutions, we can see that the spatial correlation does perform an important role.
- Using larger than 3x3 filters or deeper than 2-layer networks in this case does not help performance. The reason can be that spatial correlation happens within a small range, and more parameters in this case are not useful but harder to train.

Future Directions

- Use fully connected layer to map air quality station measures to the input matrix.
- Use convolution and deconvolution layers before and after the ConvLSTM layers to reduce the computation cost.

References

[1] X. Shi, Z. Chen, H. Wang, D.-Y. Yeung, W.-k. Wong and W. Wang-chun, Convolutional LSTM Network: A Machine Learning Approach for Precipitation Nowcasting, NIPS, 2015.