An Attention-Based Air Quality Forecasting Method

Bo Liu[1,2], Shuo Yan[2], Jianqiang Li[2], Guangzhi Qu[3], Yong Li[2], Jianlei Lang[4,5], Rentao Gu[6]
[1]Beijing Advanced Innovation Center for Future Internet Technology, Beijing University of Technology, Beijing 100124, China
[2]School of Software Engineering, Faculty of Information Technology, Beijing University of Technology, Beijing 100124, China
[3]Computer Science and Engineering Dept., Oakland University, Rochester, MI 48309, USA
[4]Key Laboratory of Beijing on Regional Air Pollution Control, Beijing University of Technology, Beijing 100124, China
[5]College of Environmental & Energy Engineering, Beijing University of Technology, Beijing 100124, China
[6]Beijing Laboratory of Advanced Information Networks, School of Information and Communication Engineering, Beijing University of Posts and Telecommunications, Beijing 100876, China
boliu@bjut.edu.cn, yanshuok@163.com, lijianqiang@bjut.edu.cn, gqu@oakland.edu, li.yong@bjut.edu.cn, jllang@bjut.edu.cn, rentaogu@bupt.edu.cn

***Abstract*—Air pollution is threatening human's health since the industrial revolution, but there are not efficient ways to solve air pollution, so forecasting air quality has become an efficient measure to prevent citizens from hurting of heavy air pollution. In this paper, we proposed an advanced Seq2Seq (Sequence to Sequence) model called attention-based air quality forecasting model (ABAFM) whose RNN encoder is replaced by pure attention mechanism with position embedding. This improvement not only reduces the training time of Seq2Seq model with attention but also enhances the robustness of Seq2Seq models. We implemented ABAFM in Olympic center and Dongsi monitoring stations in Beijing to forecast PM2.5 in future 24 hours. The experimental results showed that the proposed model outperformed the related arts, especially in sudden changes.**

Keywords—Air quality, forecasting, Seq2Seq, Attention

I. INTRODUCTION

Air pollution is a serious problem since mankind entered industrial age because of the burning of fossil fuels. In recent years, the process of industrialization has accelerated in developing countries, mankind is facing more serious air pollution problem especially in urban area. So far, most people in the world are living in cities, therefore, lots of people are suffering from air pollution. Among the pollutants, PM2.5 attracts most the attention, because it is considered as the culprit of haze. PM2.5 can be inhaled into easily and cause lung and heart diseases, such as lung cancer, myocardial infarction, etc. Therefore how to prevent people from diseases caused by air pollution is highly concerned by the governments. However, there is no efficient solution for air pollution except reducing burning fossil fuels with the current level of technology, but people have no efficient substitution of fossil fuels. At present, forecasting air quality is the most effective solution to prevent people from air pollution. With air quality forecasting, people can make protective measures in advance.

The most difficult matter of air quality forecasting is to model the complex non-linear between air pollutions and its influencing factors. Therefore, forecasting air quality is a hard task, the earliest method applied to air quality forecasting is numerical method, which use physics and chemical equations to simulate the pollutant production and spread. Theoretically, numerical method will provide a powerful forecasting, but in practice, numerical method not only cannot model every detail on pollutant production and spread but also ignores much information. Moreover, the parameters of numerical model cannot be calculated accurately, and it takes a long time to obtain result. Because of informatization, a data-driven method called statistical method was more and more popular in air quality forecasting. Statistical method utilize historical data to forecast air quality, it is easy to implement and forecast pretty fast. Multiple linear regression (MLR)[1] is a classical statistical method, some researchers use MLR to forecast air quality and get good results. However, most relationships between air quality and its influencing factors are non-linear, so some researchers tried non-linear methods such as artificial neural networks (ANNs)[2, 3], support vector machine (SVM)[4, 5], random forest (RF)[6]. Non-linear methods usually achieve better result than linear models. As the computation ability improvement, deep learning which can approximate more complex non-linearity relationship becomes popular. Li et al[7] applied an stacked auto-encoder (SAE) architecture on air quality forecasting. SAE cannot capture the sequence relationship among air quality influencing factors, Freeman et al[8], utilized long short-term memory (LSTM) to capture the sequence relationships and got a good result. In air quality forecasting for a future range, the sequence relationship exists not only in inputs but also in outputs. Vikramet al [9]used a sequence to sequence (Seq2Seq) model which is popular in machine translation to address the problem above, furthermore, Tien-Cuong et al [10] fully exploit the encoder hidden states by taking the mean of encoder hidden states as the context vector rather than final encoder hidden state.

In this paper, we proposed an attention-based air forecasting model (ABAFM). Taking the mean of encoder hidden states is not reasonable, because there are different degrees of influence to decoder stage in different time steps, attention mechanism solves this problem by weighting encoder hidden state. Moreover, recurrent neural networks (RNN) is not efficient because it cannot be parallelized, to solve this problem, Jonas et al[11] replaced RNN with convolutional neural networks (CNN), Ashish et al[12] used a self-attention mechanism to replace RNN. In our research, we adopted pure attention mechanism with position embedding. Sequential relationship cannot be well-modeled by pure attention mechanism even with position embedding, however, there is much stronger sequence relationship in decoding stage than encoding stage, so the RNN in decoder is perserved.

Our method is a kind of Seq2Seq model and it has 2 advantages than current models air quality forecasting models. One is that we applied attention to weight encoder hidden state, it is more reasonable than taking the mean of encoder hidden state. Another is adopting pure attention mechanism proposed by us makes it more efficient than traditional seq2seq models.

The structure of the rest paper is organized as follows. Section 2 introduces the proposed method in detail. Section 3 shows the performance of our proposed model and compared with other models. Section 4 is conclusion and our future work.

II. PRELIMINARIES

In this section, LSTM, GRU and Seq2Seq are introduced. LSTM and GRU are used as the decoder of ABAFM. Seq2Seq is the basic framework of ABAFM.

A. Long Short-Term Memory

LSTM[13] which was proposed by Jürgen Schmidhuber in 1997 greatly reduces the gradient vanishing in RNN and gives RNN a much longer dependency than before. At present, LSTM is so popular that RNNs adopts LSTM units are called LSTM. LSTM unit is consists of a cell, an input gate, a forget gate and an output gate. Intuitively, the cell is responsible for memory, the input gate controls the rate of new cell state are input into cell, the forget gate controls the rate of old cell state remain in cell, the output gate controls the rate of cell values are used to compute hidden state. The equations of LSTM are shown below:

$$i_t = \sigma(W_i * [h_{t-1}, x_t] + b_i) \quad (1)$$

$$f_t = \sigma(W_f * [h_{t-1}, x_t] + b_f) \quad (2)$$

$$o_t = \sigma(W_o * [h_{t-1}, x_t] + b_o) \quad (3)$$

$$\tilde{C_t} = \tanh(W_c * [h_{t-1}, x_t] + b_C) \quad (4)$$

$$C_t = f_t * C_{t-1} + i_t * \tilde{C_t} \quad (5)$$

$$h_t = o_t * \tanh(C_t) \quad (6)$$

where i_t represents the input gate; f_t represents the forget gate; o_t represents the output gate; W and b with different subscripts represent weights and biases of different linear transformation; $\tilde{C}$ is the candidate state of cell; C is the cell state; h is the hidden state of LSTM. Subscript t represent the current time step, $t-1$ is the previous time step; x is the input. $\tanh(\cdot)$ is defined as :

$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (7)$$

B. Gated Recurrent Unit

GRU (Gated Recurrent Unit) [14] is a variant of LSTM, it is simpler than LSTM but sometimes even more powerful than LSTM. Compared with LSTM, GRU consists of a cell, an update gate, and a reset gate. The responsibility of GRU cell is same as LSTM cell, and the cell state is merged with hidden state, the update gate controls the weight of candidate state and previous hidden state when calculating current hidden state, the reset gate controls the importance of previous hidden state to candidate state. The equations of GRU are shown below:

$$u_t = \sigma(W_u * [h_{t-1}, x_t] + b_u) \quad (8)$$

$$r_t = \sigma(W_r * [h_{t-1}, x_t] + b_r) \quad (9)$$

$$\tilde{h}_t = \tanh(W_h * [r_t * h_{t-1}, x_t] + b_h) \quad (10)$$

$$h_t = (1 - u_t) * h_{t-1} + u_t * \tilde{h}_t \quad (11)$$

where u_t represents the update gate; r_t represents the reset gate; $\tilde{h}$ is the candidate state; h is the cell state as well as hidden state. Other symbols have very similar meanings to the symbols in LSTM.

C. Sequence to Sequence

Seq2Seq [15, 16] is a method that adopts encoder-decoder architecture. Seq2Seq encodes source sequence into a context vector and decode the context vector into target sequence. In general, Seq2Seq consists of a RNN encoder and a RNN decoder, the structure of Seq2Seq is shown in Fig 1.

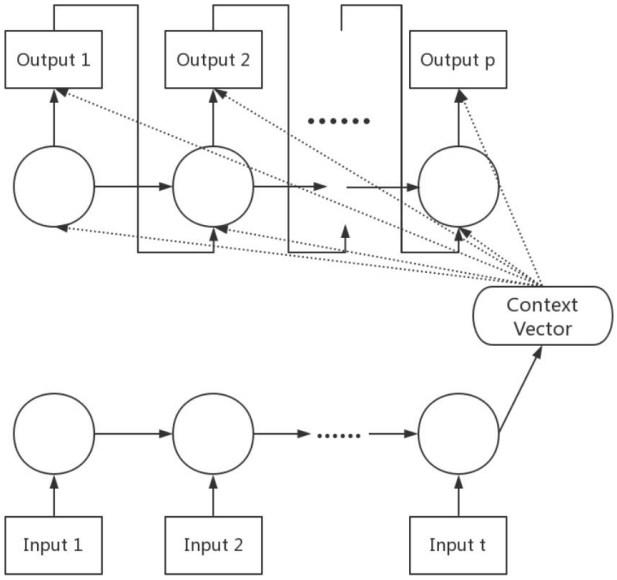

Fig. 1. The encoder-decoder architecture of Seq2Seq. The input is source sequence, the output is forecasting value.

The encoder is a standard RNN, it receives data at each time step and produces a context vector which is the last hidden state of encoder. The decoder takes the forecasting value of previous time step and the context vector as input. Take GRU for example, the update gate, reset gate and candidate state of GRU decoder are changed to:

$$u_t = \sigma(W_u * [h_{t-1}, x_t, c] + b_u) \quad (12)$$

$$r_t = \sigma(W_r * [h_{t-1}, x_t, c] + b_r) \quad (13)$$

$$\tilde{h}_t = \tanh(W_h * [r_t * h_{t-1}, x_t, c] + b_h) \quad (14)$$

where x_t is the input data, c is context vector. The change is same when regard LSTM as the decoder. In the decoder, an immediate results of forecasting value are used to predict the result of next time step, therefore, seq2seq can take advantage of sequence relationship among predictions.

III. METHODOLOGY

Vanilla Seq2Seq has two problems, one is the context vector of original Seq2Seq cannot preserve whole information of encoder sequence, another one is the RNN encoder connot be paralleled. An easy way to solve the former problem is taking the mean of every encoder hidden state as context vector. However, in practice, a forecasting

value may highly relate to some encoder hidden states. A reasonable way to fix this problem is attention mechanism which was first introduced into NMT by Bahdanau et al[17] in 2014 and then perfected by Luong et al[18] in 2015. Attention mechanism can give higher weights to the encoder hidden state that highly relates to forecasting value. The latter problem can be dealt by replacing RNN decoder with pure attention mechanism which can parallelize encoder process.

Pure attention mechanism weights the source sequence by measuring the score between current decoder hidden state and each time step of source sequence. Replacing RNN with pure attention mechanism cannot fully obtain the sequential information of source sequence, so we applied position embedding. The position embedding embeds the absolute position information into a trainable parameter matrix $PE = \{p_1, p_2, \dots, p_t\}$, PE has the same dimension as source sequence. So the score function of ABAFM is defined by:

$$\text{score}(h_t, x_s) = h_t^T tanh(W(x_s + PE)) \tag{15}$$

where h_t is decoder hidden state; W is a weight matrix which is shared in different time step. By using pure attention mechanism, the score function for every x_s can be computed at the same time. To insure the summation of weights is 1, a softmax function is used to normalize these weights:

$$a_t(s) = \frac{\exp\{\text{score}(h_t, \bar{h}_s)\}}{\sum_{s'} \exp\{\text{score}(h_t, \bar{h}_{s'})\}} \tag{16}$$

where a_t denotes the weight matrix. Then, a_t element-wise multiply by each encoder state and get the context vector c_t. c_t and h_t are used to obtain the forecasting value of air quality :

$$p_t = W[c_t, h_t] \tag{17}$$

where p_t is the forecasting value of air quality at t th time step.

IV. Experiment Settings

A. Dataset

In this paper, we utilized the hourly air quality data and hourly weather data of Beijing from April 2017 to March 2018. The air quality data contain SO_2, O_3, NO_2, CO, PM2.5. The weather data contain temperature, humidity, wind force, wind direction and precipitation.

B. Features and output

For Seq2Seq model, We took past 72 hours air quality data and weather data as encoder input, the decoder input consists of air quality data of previous time step, and the real value of weather forecast, so decoder input can be represented as follows:

$$X_{decoder} = [X_{weather}, p_{t-1}]$$

where p_{t-1} denotes the forecasting value of previous time step, it consists of the concentration of SO_2, O_3, NO_2, CO, and PM2.5, but only PM2.5 is our target. The decoder input of first time step is the air quality data of previous hour and the current weather forecast data.

C. Evaluation

In this paper, we utilize root mean squared error (RMSE) and R^2 as performance metrics. RMSE is defined as follows:

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=0}^{n}(O_i - P_i)^2} \tag{18}$$

Where O denotes the observation value, P denotes the forecasting value, n denotes the number of samples. R^2 is defined as :

$$R^2 = 1 - \frac{\sum_{i=0}^{n}(O_i - P_i)^2}{\sum_{i=0}^{n}(O_i - \bar{O})^2} \tag{19}$$

Where $\bar{O}$ is the mean of observation values.

V. Result And Discussion

In this paper, we took the data from April 2017 to February 2018 as training data and evaluate on the data from March 1 2018 to March 7 2018. Among 35 air quality monitor station, Dongsi and Olympic center station were chosen as the target station, because they are populous. Except for proposed method, Seq2Seq, Seq2Seq-Mean and Seq2Seq-attention were chosen as baseline. The context vector of Seq2Seq-mean is mean of encoder hidden state. Seq2Seq-attention obtains context vector by attention mechanism. Our experiment was carried out on a computer with I7 6770HQ, 16GB RAM and GTX1060 6GB.

TABLE I. RMSE Of Different Methods and Stations

Stations	Methods	4	8	12	16	20	24	Total
Olympic Center	Seq2Seq(GRU)	29.210	39.175	43.718	45.304	48.461	50.275	43.259
	Seq2Seq(LSTM)	34.080	34.365	36.015	38.821	44.044	51.214	40.227
	Seq2Seq-Mean(GRU)	28.241	35.907	39.513	45.353	50.574	55.951	43.576
	Seq2Seq-Mean(LSTM)	37.463	40.390	43.120	44.392	46.729	50.623	43.991
	Seq2Seq-Attention(GRU)	27.990	33.098	40.773	46.530	49.032	53.250	42.712
	Seq2Seq-Attention(LSTM)	31.190	31.916	34.402	39.053	45.943	53.098	40.063
	ABAFM (GRU)	25.225	**27.968**	**27.149**	**31.417**	**33.542**	**35.053**	**30.266**
	ABAFM (LSTM)	**24.169**	35.712	39.613	43.359	42.075	40.492	38.119
Dongsi	Seq2Seq(GRU)	47.966	61.014	58.598	60.179	63.957	70.900	60.822
	Seq2Seq(LSTM)	65.598	79.017	78.434	76.741	72.593	80.290	75.614
	Seq2Seq-Mean(GRU)	42.149	51.263	58.017	61.441	63.861	68.772	58.243
	Seq2Seq-Mean(LSTM)	62.071	74.096	75.451	76.373	70.316	75.456	72.464
	Seq2Seq-Attention(GRU)	45.025	48.592	52.689	**54.470**	**59.262**	**64.514**	**54.476**
	Seq2Seq-Attention(LSTM)	56.737	68.640	69.765	78.328	72.161	74.873	70.411
	ABAFM (GRU)	**40.656**	**43.779**	**49.250**	57.615	64.658	68.308	55.017
	ABAFM (LSTM)	44.628	54.335	65.278	62.508	61.925	65.572	59.508

TABLE II. R^2 OF DIFFERENT METHODS AND STATIONS

Stations	Methods	4	8	12	16	20	24	Total
Olympic Center	Seq2Seq(GRU)	0.710	0.472	0.340	0.284	0.169	0.097	0.345
	Seq2Seq(LSTM)	0.607	0.594	0.553	0.475	0.314	0.062	0.434
	Seq2Seq-Mean(GRU)	0.730	0.557	0.462	0.283	0.095	-0.118	0.334
	Seq2Seq-Mean(LSTM)	0.525	0.439	0.358	0.313	0.227	0.084	0.324
	Seq2Seq-Attention(GRU)	0.734	0.623	0.426	0.245	0.149	-0.013	0.361
	Seq2Seq-Attention(LSTM)	0.671	0.650	0.592	0.468	0.253	-0.007	0.438
	ABAFM (GRU)	0.784	**0.731**	**0.746**	**0.656**	**0.602**	**0.561**	**0.680**
	ABAFM (LSTM)	**0.802**	0.562	0.459	0.345	0.374	0.414	0.493
Dongsi	Seq2Seq(GRU)	0.596	0.333	0.369	0.317	0.211	0.020	0.308
	Seq2Seq(LSTM)	0.245	-0.118	-0.129	-0.110	-0.015	-0.256	-0.064
	Seq2Seq-Mean(GRU)	0.688	0.528	0.382	0.288	0.214	0.078	0.363
	Seq2Seq-Mean(LSTM)	0.324	0.016	-0.045	-0.099	0.047	-0.109	0.022
	Seq2Seq-Attention(GRU)	0.644	0.576	0.490	**0.440**	**0.322**	**0.188**	**0.443**
	Seq2Seq-Attention(LSTM)	0.435	0.155	0.106	-0.157	-0.003	-0.092	0.074
	ABAFM (GRU)	**0.710**	**0.656**	**0.554**	0.374	0.194	0.091	0.430
	ABAFM (LSTM)	0.651	0.471	0.218	0.263	0.261	0.162	0.337

A. *Experiment Result*

Tables I-II show the RMSE and R^2 every 4 hours of the Olympic center station and Dongsi station, the last column is the total result of 24 hours. From these tables we can see that the performance all these methods get worse as time goes on because the forecasting error is accumulating and the correlation between forecasting values and past observation values is decreasing. GRU ABAFM has the best result on Olympic center station without doubt. For Dongsi station, GRU Seq2Seq-Attention has the best result in terms of total result. But if we look in detail, GRU ABAFM has better performance than GRU Seq2Seq-Attention in previous 12 hours, the forecasting values of previous hours is much more believable than the last 12 hours, the LSTM ABAFM is just a little better than GRU ABAFM. In a word GRU ABAFM has the best performance on Dongsi station.

The RMSE and R^2 of GRU Seq2Seq in Olympic center is improved slightly by using mean of encoder hidden states, but the performance of LSTM Seq2Seq in Olympic center gets worse. In Dongsi station, utilizing mean of encoder hidden states really help Seq2Seq impove its performance. The forecasting result of GRU Seq2Seq is improved obviously by attention in Dongsi and Olympic center. The LSTM Seq2Seq cannot achieve desired performance but still get imporved by attention in both stations. In a word, utilizing the mean of encoder hidden states improve the performance of Seq2Seq, but taking mean of encoder hidden states is not reasonable, so attention mechanism which enforces the influence of useful factors and reduce the influence of useless factors get better performance than Seq2Seq-Mean. ABAFM has even better performance than Seq2Seq-Attention and gives Seq2Seq an obvious improvement proves that cancelling the RNN encoder is a good choice. According to our experiment, we can see that GRU performs better than LSTM in each station with each method, GRU may be a better choice than LSTM for Seq2Seq in air quality forecasting. Attention mechanism can improve the performance of previous hours significantly, maybe this is because it captures more relationship between current time step and past sequence, as the relationship decreased, the performance of attention mechanism also decreased but still perserve. ABAFM perform good especially in previous hours, because pure attention mechanism capture the relationship between current time step and original data directly.

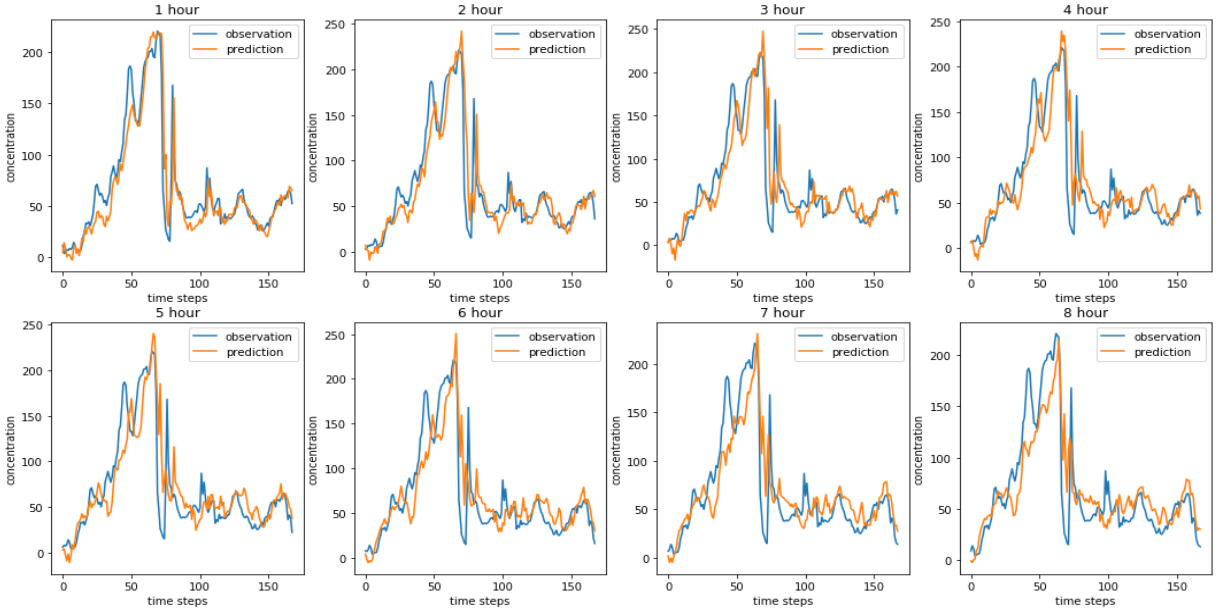

Fig. 2. Olympic center result of previous 8 hours

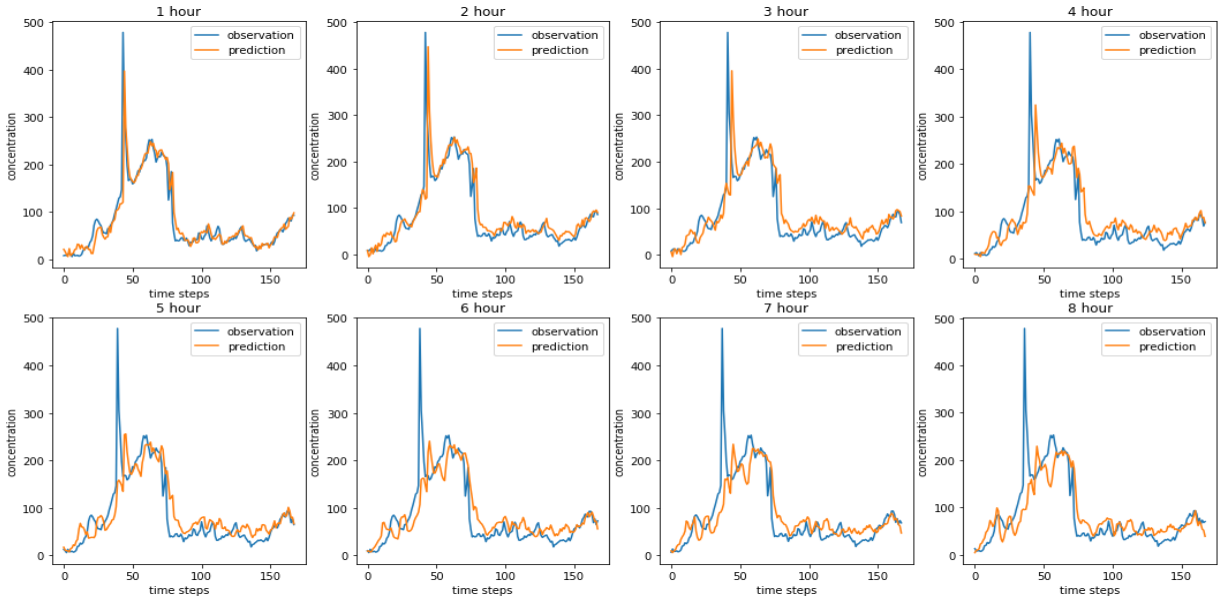

Fig. 3. Dongsi result of previous 8 hours

TABLE III. TIMES

Stations	Methods	Training Time	Forecasting Time
Olympic Center	Seq2Seq(GRU)	**392.832**	3.647
	Seq2Seq(LSTM)	399.888	3.609
	Seq2Seq-Mean(GRU)	509.184	**1.931**
	Seq2Seq-Mean(LSTM)	501.236	2.034
	Seq2Seq-Attention(GRU)	1545.841	4.104
	Seq2Seq-Attention(LSTM)	1537.344	3.793
	ABAFM (GRU)	1007.712	2.563
	ABAFM (LSTM)	997.776	2.422
Dongsi	Seq2Seq(GRU)	**371.520**	3.294
	Seq2Seq(LSTM)	374.976	3.362
	Seq2Seq-Mean(GRU)	479.281	**1.899**
	Seq2Seq-Mean(LSTM)	472.865	2.133
	Seq2Seq-Attention(GRU)	1605.728	3.608
	Seq2Seq-Attention(LSTM)	1560.384	3.604
	ABAFM (GRU)	1030.176	2.276
	ABAFM (LSTM)	967.824	2.586

B. *Times*

Table III shows the training time and forecasting time of each model, we can see that Seq2Seq has the shortest training time, Seq2Seq with attention has the longest training time because attention mechanism is the most time-consuming part. ABAFM has much shorter training time than Seq2Seq with attention due the RNN encoder is replaced by pure attention mechanism. By applying pure attention mechanism, about 1/3 time is saved when training Seq2Seq model. The forecasting time of all models is within 5 seconds, in practice, these models can be used to real-time forecasting. In summary, replacing RNN encoder with pure attention mechanism can reduce the training time.

C. *Visualization*

The forecasting results in 8 hours of GRU ABAFM on the 2 stations are shown in Figures 2-3.

From Fig. 2 we can see that the forecasting values of previous 4 hours is very close to the observation values, even near the peak region and valley region, but start from 5th hour, the errors of peak region and valley region become larger. In Fig. 3, we can see that Dongsi station has a much tougher curve than Olympic center, the model forecasts extremely well in the previous 2 hours and it is acceptable in previous 4 hours, start from 5th hours, the model cannot predict sudden pollution well. Together the two pictures we can conclude that our model can forecast sudden pollution 4 hours in advance.

VI. CONCLUSION AND FUTURE WORK

In this paper, we proposed an attention-based air quality forecasting model. By replacing the RNN encoder with pure attention mechanism and position embedding, our experiments proved that our model is more efficient on training and more accurate on air quality forecasting. We also found that attention can help Seq2Seq improve performance and GRU is more suitable for Seq2Seq on air quality forecasting. Our model can forecast sudden pollution 4 hours in advance, so that people have enough time to take actions to prevent themselves from air pollution. In the future, we will further study forecasting sudden pollution earlier and improve the forecasting accuracy in longer time.

ACKNOWLEDGMENT

This work is supported by National Natural Science Foundation of China (61702021), Beijing Natural Science Foundation (4174082), General Program of Science and Technology Plans of Beijing Education Committee (SQKM201710005021), Fundamental Research Foundation of Beijing University of Technology (PXM2017_014204_500087), and Funds of Beijing Advanced Innovation Center for Future Internet Technology of Beijing University of Technology (BJUT).

REFERENCES

[1] Rajput, T.S., and Sharma, N.: Multivariate regression analysis of air quality index for Hyderabad city: Forecasting model with hourly frequency, 2017

[2] Niu, Y.X.: Research on Air Quality Prediction Model Based on Genetic Algorithm and BP Neural Network, Computer Engineering & Software, 2017

[3] Corani, G.: Air quality prediction in Milan: feed-forward neural networks, pruned neural networks and lazy learning, ECOL MODEL, 2005, 185, (2-4), pp. 513-529

[4] Li, Z.H., and Yang, J.: PM-25 forecasting use reconstruct phase space LS-SVM, in Editor (Ed.) PM-25 forecasting use reconstruct phase space LS-SVM (2010,edn.), pp. 143-146

[5] Yin, Q., Hong-Ping, H.U., Bai, Y.P., Wang, J.Z., and Science, S.O.: Prediction of Air Quality Index in Taiyuan City Based on GA-SVM, Mathematics in Practice & Theory, 2017

[6] Hou, J., Qi, L.I., Zhu, Y., Feng, X., and Mao, X.: Real-time forecasting system of PM2.5 concentration based on spark framework and random forest model, Science of Surveying & Mapping, 2017

[7] Li, X., Peng, L., Hu, Y., Shao, J., and Chi, T.: Deep learning architecture for air quality predictions, Environmental Science & Pollution Research, 2016, 23, (22), pp. 22408-22417

[8] Freeman, B.S., Taylor, G., Gharabaghi, B., and Thé, J.: Forecasting air quality time series using deep learning, J AIR WASTE MANAGE, 2017, (2)

[9] Reddy, V., Yedavalli, P., Mohanty, S., and Nakhat, U.: Deep Air: Forecasting Air Pollution in Beijing, China

[10] Bui, T.C., Le, V.D., and Cha, S.K.: A Deep Learning Approach for Forecasting Air Pollution in South Korea Using LSTM, 2018

[11] Gehring, J., Auli, M., Grangier, D., Yarats, D., and Dauphin, Y.N.: Convolutional Sequence to Sequence Learning, 2017

[12] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., and Polosukhin, I.: Attention Is All You Need, 2017

[13] Lipton, Z.C., Berkowitz, J., and Elkan, C.: A Critical Review of Recurrent Neural Networks for Sequence Learning, Computer Science, 2015

[14] Dey, R., and Salemt, F.M.: Gate-variants of Gated Recurrent Unit (GRU) neural networks, in Editor (Ed.) Gate-variants of Gated Recurrent Unit (GRU) neural networks (2017,edn.), pp. 1597-1600

[15] Cho, K., Van Merrienboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y.: Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation, Computer Science, 2014

[16] Sutskever, I., Vinyals, O., and Le, Q.V.: Sequence to Sequence Learning with Neural Networks, 2014, 4, pp. 3104-3112

[17] Bahdanau, D., Cho, K., and Bengio, Y.: Neural Machine Translation by Jointly Learning to Align and Translate, Computer Science, 2014

[18] Luong, M.T., Pham, H., and Manning, C.D.: Effective Approaches to Attention-based Neural Machine Translation, Computer Science, 2015Sutskever, I., Vinyals, O., and Le, Q.V.: Sequence to Sequence Learning with Neural Networks, 2014, 4, pp. 3104-3112