基于 RNN 的空气污染时空预报模型研究

范骏翔[1]，李　琦[1,2]，朱亚杰[1]，侯俊雄[1]，冯　逊[1]

（1. 北京大学 遥感与地理信息系统研究所，北京　100871；
2. 北京大学 智慧城市研究中心，北京　100871）

摘　要：针对空气污染物时间序列中包含缺失值以及现有时间序列预报模型缺乏对时序特征状态建模的问题，该文构建了基于缺失值处理算法和 RNN(循环神经网络)的时空预报框架。对空气污染物时序数据设计了 3 种缺失值处理算法(前向递补、均值替代和权重衰减)，用缺失标签和缺失时长对缺失值建模，并在此基础上搭建含有全连接层与 LSTM 层的深度循环神经网络(DRNN)用于时空预报。使用深度全连接神经网络(DFNN)作为 DRNN 的对照，用京津冀区域的空气质量和气象数据训练模型，并比较不同模型的预测精度。通过实验，比较了 3 种缺失值处理方法的效果，结果表明，LSTM 在空气污染时空序列预测上的表现优于传统的全连接神经网络层，证实了提出的基于深度学习的时空预报框架的有效性。

关键词：空气污染；缺失值；RNN；LSTM；深度学习

【中图分类号】P208　　**【文献标志码】**A　　**【文章编号】**1009-2307(2017)07-0076-08

DOI：10.16251/j.cnki.1009-2307.2017.07.013

Aspatio-temporal prediction framework for air pollution based on deep RNN

Abstract：Time series data in practical applications always contain missing values. In order to handle missing values in time series, as well as the lack of considering temporal attributes in current time series prediction models, we develop a temporal-spatio prediction framework based on missing value processing algorithms and RNN(Recurrent Neural Network). In this paper, three different missing value modelling algorithms are implemented by using missing tag and missing interval to represent missing pattern in time series data. On top of the missing value modelling algorithms, we construct a deep neural network with LSTM layers and fully connected layers to perform prediction tasks. Real-world air quality and meteorological datasets(Jingjinji Area)are used to train different kinds of deep neural networks, in which the deep feed forward neural networks serve as baseline models. Performances of different models are evaluated in order to compare capabilities of three missing value modelling algorithms, as well as prediction accuracy of different neural network architectures. Experiment results show that deep neural with LSTM layers perform better than those with only fully connected layers, and validate the capability of the suggested temporal-spatio prediction framework based on deep learning.

Keywords: air pollution; missing value; RNN; LSTM; deep learning

FAN Junxiang[1], *LI Qi*[1,2], *ZHU Yajie*[1], *HOU Junxiong*[1], *FENG Xiao*[1] (1. Peking University School of Earth and Space Sciences, Beijing 100871, China; 2. Peking University Smart City Research Center, Beijing 100871, China)

作者简介：范骏翔(1992—)，男，湖北武汉人，硕士研究生，主要研究方向为智慧城市、交通与空气污染预测。
E-mail：junxiang.fan@pku.edu.cn

收稿日期：2017-04-01
基金项目：国家科技支撑计划项目(2012BAC20B06)

通信作者：李琦　教授 E-mail：qi.lee009@gmail.com

0　引言

根据世界卫生组织的估计[1]，全世界每年有近 200 万人的死亡与空气污染相关，其中约有半数来自发展中国家。随着城市化的推进，工业化程度和机动车保有量逐年上升。机动车数量增长不仅带来拥堵，也排放大量污染物。各类空气污染物中受到关注程度较高的是 PM2.5，即粒径小于 2.5 μm的颗粒。虽然 PM2.5 与呼吸道和心脑血管

疾病的详细病理相关的研究还在进行中，但已经有文献研究表明，PM2.5 与一些疾病的死亡率存在显著的正相关关系[2-3]。

空气污染物模型可以分为两类：①机理模型，即基于污染物的生成与传输机理，追踪和模拟污染物变动，所得结果易于理解与分析；②统计学习模型，无须深入了解污染物物理化学性质，直接从数据中发掘潜在的模式，其结果无法给出机理解释。

第一类模型包括文献［4］提出的 CMAQ 模型和文献［5］提出的 WRF/Chem 模型，分别对污染物传输过程进行“离线”和“在线”模拟。第二类模型中包括线性回归、地理加权回归(GWR)、土地利用回归(LUR)、SVM(支持向量机)、ANN(神经网络，FNN)等。文献［6］用 LUR 分析了欧洲的 PM2.5 和 PM10 分布。文献［7］通过 GWR 模型，对全中国 PM2.5 的浓度分布作天级别的预测。文献［8］使用 PCA(主成分分析)分析空气质量/气象因素对 PM2.5 的影响，并用 ANN 预测 PM2.5 分布。文献［9］使用小波变换加 SVR(支持向量回归)，预测 PM2.5 浓度。循环神经网络(recurrent neural network，RNN)是 ANN 的一种变体，可以模拟序列数据内在的依赖关系，因此广泛用于自然语言处理、图片标注和机器翻译等领域，在空气质量研究中使用较少。文献［10］(2013)比较了 RNN 和 ARIMA 的时间序列预测效果，并结合二者预测污染物分布。文献［11］通过在 RNN 中添加动态 autoencoder 层，提升了 PM2.5 预测精度。文献［12］使用 RNN，基于出租车 GPS 数据，预测交通网络的拥堵情况。文献［13］设计了基于 GRU(gated recurrent unit)的深度网络 GRU-D，能有效处理缺失数据，同时对医疗数据做出预测。

时间序列数据常常会出现缺失值，可能原因包括设备故障、网络中断、异常值等等，因此如何处理缺失值，是时间序列相关研究中的一个重要的问题。一种常用方法是忽略序列中的缺失值，直接基于观测到的连续数据建模。也有研究尝试对缺失值进行补充，比如平滑、插值、核方法等[14-16]。这些方法也有一些缺陷，即建模之前需要拟合缺失值的分布，而且没有利用缺失数据自身所包含的时间序列信息。基于时间序列缺失值的建模研究在医疗信息预测，地震研究等领域应用较多，在空气质量时序预报方面应用相对较少。

基于以上的理论基础，本文试图建立针对含缺失值的污染物时间序列的预测模型，主要工作包括：① 通过设置缺失标签/缺失时长，研究缺失值分布，并设计 3 种时间序列缺失值处理算法；② 在时间序列缺失值处理算法的基础上，分别结合 RNN 和 FNN 搭建了时序预报框架；③实现了基于 LSTM 的深度神经网络，并与全连接深度神经网络比较精度。

1　关键模型与算法

1.1　问题的形式化定义

对于 N 个不同的监测站点，获取到的监测数据可以用 N 个时间序列表示，组成集合 $\mathrm{ST}=\{\mathrm{st}_1,\mathrm{st}_2,\cdots,\mathrm{st}_N\}$。$st_n=\{X_1,X_2,\cdots,X_T\}$ $(n=1,\cdots,N)$表示单个站点 n 的监测数据序列(共 T 个时间步)，序列中的每个观测值 X_t都是一个 d 维向量。本研究中的问题可以定义为：在给定的时刻 t，基于目标站点 st_n 找到一个全站点集合 ST 的最优子集 $\mathrm{ST}_{\mathrm{sub}}$，使用 $\mathrm{st}_n\cup\mathrm{ST}_{\mathrm{sub}}$ 中时刻 $(t,t-1,\cdots,t-L)$的历史数据，预测 st_n 在时刻 $(t+1,t+2,\cdots,t+F)$的对应值。最优子集可以由当前站点的最近邻站点构成。具体而言，模型的输入数据是多站点的历史数据，包括空气质量数据和气象数据两部分，输出结果是多站点的 PM2.5 浓度预测值。

1.2　缺失值处理算法

缺失值处理算法的核心是两个变量：缺失标签和缺失时长。

1)缺失标签。对于一个有缺失的时间序列 st_n，用 s_t 来表示第 t 个时间步对应的时刻($s_1=0$)。对于观测量 $\boldsymbol{X}_t\in\boldsymbol{R}^d$，定义向量 $\boldsymbol{m}_t\in\{0,1\}^d$ 表示其值缺失情况，其每个维度 m_t^d 都是缺失标签，如式(1)所示。

$$m_t^d=\begin{cases}1, & x_t^d\text{ 有效}\\ 0, & x_t^d\text{ 缺失}\end{cases}\tag{1}$$

2)缺失时长。基于缺失标签 m_t^d 以及时刻 s_t，就可以定义观测值的某一维度上数据的缺失时长 δ_t^d，即该维度从最近一次观测到有效值，到当前时刻所经过的时长，如式(2)所示。

$$\delta_t^d=\begin{cases}s_t-s_{t-1}, & t>1,m_t^d=1\\ \delta_{t-1}^d+s_t-s_{t-1}, & t>1,m_t^d=0\\ 0, & t=1\end{cases}\tag{2}$$

在不改动神经网络结构的基础上，本研究中设计并应用了 3 种面向原始数据的缺失值处理办法。

1) 用最近的有效数据替代缺失值(前向递

补)，如式(3)所示。

$$x_t^d \leftarrow m_t^d x_t^d + (1-m_t^d)x_{t'}^d \tag{3}$$

式中：$t'(<t)$表示 d 维分量最近被观测到的时间步；$x_{t'}^d$ 表示该分量最近有效的观测值。

2) 用相同时间段的数据均值替代缺失值(均值替代)，如式(4)所示。

$$x_t^d \leftarrow m_t^d x_t^d + (1-m_t^d)\widetilde{x}^d \tag{4}$$

式中：$\widetilde{x}^d$ 表示观测量的第 d 维分量在同一个月中当前时刻的均值。

$$\widetilde{x}^d = \sum_{t'=1}^{T} m_t^d x_t^d I_{\text{hour}}(s_t, s_{t'}) / \sum_{t'=1}^{T} m_t^d I_{\text{hour}}(s_t, s_{t'}) \tag{5}$$

$$I_{\text{hour}}(s_t, s_{t'}) = \begin{cases} 1, & s_t = s_{t'} \\ 0, & s_t \neq s_{t'} \end{cases} \tag{6}$$

3) 综合 1)和 2)做加权和，替代缺失值(权重衰减)。该方法的逻辑在于对于某一个观测值分量，一方面，长期来看可能存在一个稳定值；另一方面，则会受到突发状况的影响，产生波动。因此可以考虑对最近有效值(代表突发状况)和均值(代表长期稳定值)作加权和，用于替代缺失值。关于权重分配，本研究中使用指数衰减形式控制最近有效值对缺失值补偿结果的影响，即如果当前时间步发生缺失，那么最近有效值距离当前时间步越远，则对当前缺失补偿值的贡献越弱。如式(7)所示。

$$x_t^d \leftarrow m_t^d x_t^d + (1-m_t^d)\gamma_{x_t}^d x_{t'}^d + (1-m_t^d)(1-\gamma_{x_t}^d)\widetilde{x}^d \tag{7}$$

式中：$\gamma_{x_t}^d = 1/\exp(\delta_t^d)$。

1.3 循环神经网络与 LSTM

循环神经网络(recurrent neural network，RNN)是传统的前馈神经网络(feedforward neural network，FNN)的一个变种，二者的区别在于，FNN 的神经元仅仅通过层和层之间的连接来完成信息传递，而 RNN 在网络中引入了环状结构，即建立了神经元到自身的连接。通过这种连接，RNN 能够将上一时间点上的输入以“记忆”的形式存储在网络中，并影响下一步的网络输出。对于 FNN 而言，只能将输入通过隐含层映射到输出层，而 RNN 则可以将完整的一段历史映射到每一个输出神经元中。因此，RNN 在输入输出均为序列数据的预测问题中，有着比 FNN 更好的表现。

RNN 的预测过程与 FNN 相似，都由前向传播算法来完成。FNN 的训练过程通过后向传播算法(back propagation，BP)来实现，而 RNN 因为考虑到不同时间步之间的相互影响，因此需要在时间维上对后向传递的结果进行叠加，即时间后向传播算法(back propagation through time，BPTT)。

考虑一个基础的 RNN 模型例子：输入层(I 个神经元)，隐含层(H 个神经元)和输出层(K 个神经元)，输入数据是长度为 T 的序列 X。

RNN 的前向传播算法如式(8)～式(10)。

$$a_h^t = \sum_{i=1}^{I} w_{ih} x_i^t + \sum_{h'=1}^{H} w_{h'h} b_{h'}^{t-1} \tag{8}$$

$$b_h^t = \theta_h(a_h^t) \tag{9}$$

$$a_k^t = \sum_{h=1}^{H} w_{hk} b_h^t \tag{10}$$

式中：x_i^t 是第 i 维输入在时刻 t 的值；w_{ij} 表示神经元 i 与 j 的连接权重；a_j^t 和 b_j^t 分别表示神经元 j 在时刻 t 的输入值和激活值；θ_h 表示神经元 h 使用的激活函数。

RNN 的时间后向传播算法首先定义损失函数对神经元 j 在时刻 t 输入值的偏导数，然后通过链式法则计算损失函数对网络权重的偏导数，如式(11)所示。

$$\delta_j^t = \frac{\partial L}{\partial a_j^t} \tag{11}$$

$$\delta_h^t = \theta'(a_h^t)\left(\sum_{k=1}^{K} w_{hk}\delta_k^t + \sum_{h'=1}^{H} w_{hh'}\delta_{h'}^{t+1}\right) \tag{12}$$

$$\frac{\partial L}{\partial w_{ij}} = \sum_{t=1}^{T} \frac{\partial L}{\partial a_j^t}\frac{\partial a_j^t}{\partial w_{ij}} = \sum_{t=1}^{T} \delta_j^t b_i^t \tag{13}$$

该算法与 FNN 的不同之处在于：① 损失函数与神经元之间的关系同时受到当前时间步(t)输出层(K)和下个时间步($t+1$)隐含层(H)的影响；② 对每个时间步利用链式法则进行计算，将所有结果在时间维度上相加，得到损失函数对于神经网络权重的偏导数。通过梯度下降(gradient descent)，更新神经网络中的权重，直至收敛。

以上是 RNN 的训练过程，可以看出，RNN 能够对输入数据的序列特性进行“记忆”，但一些研究表明[17-18]：随着时间步迭代的进行，历史输入值对于隐含层的影响会逐渐削弱直至消失，这个问题被称为梯度消失问题(vanishing gradient)。

为了解决此问题，文献[19](1997)提出了长短期记忆(long short-term memory，LSTM)神经网络。LSTM 是 RNN 的一种变体，将 RNN 中隐含层的神经元替换成了记忆体(memory block)，每个记忆体中包含一到多个记忆细胞(memory cell)和 3 种非线性求和单元。非线性求和单元又被称作“门”(gate)，分为 3 种：“输入门(input gate)”“输出门(output gate)”和“遗忘门(forget gate)”，分别通过矩阵乘法控制记忆细胞的输入、

输出以及内部“状态(state)”传递。LSTM 中的消息传递包括两方面：① 跨记忆体的消息传递，只有记忆细胞的输出值会经历这个过程；②记忆体内部的消息传递，包括记忆细胞的状态值，记忆细胞的输入值，以及各个门单元的激活值等，都只在一个记忆体内部进行传递[20]。

LSTM 的前向传播算法与 RNN 类似，输入数据是一个长度为 T 的时间序列，时间步每前进一步，便更新一次输出结果。算法流程是：①计算输入门的值；②计算遗忘门的值；③计算细胞内部“状态”值；④计算输出门的值；⑤计算细胞输出值。

LSTM 的时间后向传播算法与 RNN 中的类似，从时间序列的末尾(时刻 T)开始，逐步反向循环计算各参数的梯度，最后用各时间步的梯度更新网络参数，核心计算过程是：① 计算记忆细胞输出值对应的偏导数；② 计算输出门对应的偏导数；③ 计算记忆细胞状态对应的偏导数；④ 计算遗忘门对应的偏导数；⑤计算输入门对应的偏导数；⑥用梯度下降更新权重。

1.4　算法流程与评价指标

基于上述关键模型与算法，本文搭建一个深度学习算法框架，流程主要包括 3 步：①数据预处理，对原始数据中的奇异值、编码问题等进行处理，并通过缺失值处理算法(前向递补、均值替代、权重衰减)，将含有缺失值的非连续时间序列数据完整化；②数据融合与格式化，基于数据的时间和空间分布，对多源数据(空气质量数据、气象数据、站点位置数据)进行时/空融合，并通过设置滑动时间窗口，生成用于训练和测试模型的时间序列数据；③训练与评价模型，使用②生成的输入数据，训练基于 LSTM 的深度神经网络，并使用特定的评价指标(RMSE/MAE/IA)来评估网络的预测效果。算法流程如图 1 所示。

因为本研究针对的问题类型是回归(预测空气污染物浓度的实数值)，因此评价指标使用 RMSE、MAE 以及一致性指数(index of agreement，IA)3 种。RMSE 与 MAE 分别如式(14)和式(15)所示。

$$\mathrm{RMSE}=\sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i-\hat{y}_i)^2} \tag{14}$$

$$\mathrm{MAE}=\frac{1}{n}\sum_{i=1}^{n}|y_i-\hat{y}_i| \tag{15}$$

一致性指数 IA 是由 Willmott 在 1981 年提出的一种与维度无关的，用于衡量模型预测结果平均误差的指标[21]，如式(16)所示。

$$\mathrm{IA}=1-\frac{\sum_{i=1}^{n}(P_i-O_i)^2}{\sum_{i=1}^{n}(|P_i-\bar{O}|+|O_i-\bar{O}|)^2} \tag{16}$$

式中：P_i 和 O_i 分别表示每个样本的预测值和真实值；$\bar{O}$ 表示真实值的均值。IA 的值被限制在 0～1 的范围内，表示预测值和数据真实值关于某一个“真实均值”的协变关系。IA 的值越接近 1，说明预测值和真实值的分布越相似，拟合的结果越可靠。

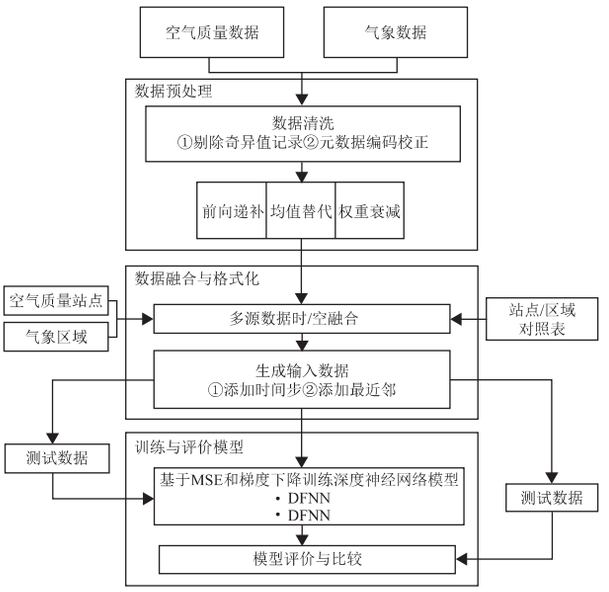

图 1　算法流程图

Fig. 1　Algorithm Flow Chart

2　实验与结果分析

2.1　研究区域与数据源

本研究所针对的区域是京津冀地区，使用的数据来自环保局和气象局网站提供的 API，数据源分为 3 类：① 空气质量监测数据；②气象监测数据；③站点数据。

空气质量监测数据每小时更新一次，包括 6 个方面数据：PM2.5 浓度、PM10 浓度、O_3 浓度、SO_2 浓度、NO_2 浓度和 CO 浓度。

气象监测数据 15～30 min 更新一次，包括 4 个方面：温度、湿度、风速和风向。

数据的时间跨度为 2014 年 6 月 1 日—2015 年 1 月 31 日，共计 7 个月。此外，站点数据使用的是京津冀地区的 80 个空气质量监测站点信息，包含每个站点的 ID 以及坐标(经纬度)，站点的位置分布如图 2 所示。

2.2　模型与结果分析

本研究中，所使用的深度神经网络主要由

图 2 京津冀地区空气质量站点分布

Fig. 2 Spatial Distribution of Air Quality Monitoring Stations in Jingjinji Area

LSTM 层和全连接层组成，输入数据是包含 19 维特征的历史时间序列，这些特征可以概括为 5 方面（见表 1）。

表 1 输入数据特征

Tab. 1 Input Features

监测量	单位
PM2.5	$\mu g/m^3$
PM10	$\mu g/m^3$
O_3	ppb
SO_2	ppb
NO_2	ppb
CO	ppb
Temperature	℃
Wind _direction	NA
Wind _speed	级
humidity	NA
weekday	NA
month	NA
day	NA
hour	NA
longitude	(°)
latitude	(°)
Nearest Neighbour1 PM2.5	$\mu g/m^3$
Nearest Neighbour 2 PM2.5	$\mu g/m^3$
Nearest Neighbour 3 PM2.5	$\mu g/m^3$

① 本地空气质量属性，包括 PM2.5 浓度、PM10 浓度、O_3 浓度、SO_2 浓度、NO_2 浓度、CO 浓度；②本地气象属性，包括温度、风向、风速、湿度；③最近邻站点空气质量属性，包括几个最近邻站点的 PM2.5 浓度；④时间属性：周中天数（weekday）、日期、月份以及小时时刻；⑤空间属性：站点的坐标（经纬度）。输出数据则是一个 1 维标量，即未来某时刻当前站点的 PM2.5 浓度。

本文使用过去 48 h 的气象以及空气质量数据预测各站点未来 1 h 的 PM2.5 浓度分布，然后对预测值进行空间插值得到京津冀区域的 PM2.5 分布情况，从而达成对于区域空气污染分布的时序预报。

设计的深度神经网络分为两种：① 基于 LSTM 的深度循环神经网络 DRNN（含有 1 层 LSTM 或 2 层 LSTM，其余为全连接层）；②完全由全连接层组成的深度前馈神经网络 DFNN（与含 1 层或 2 层 LSTM 的 DRNN 对应），框架分别如图 3(a)至图 3(d)所示。其中 DRNN 为本研究设计实现的主要模型，而 DFNN 则是 DRNN 的对照基准模型。

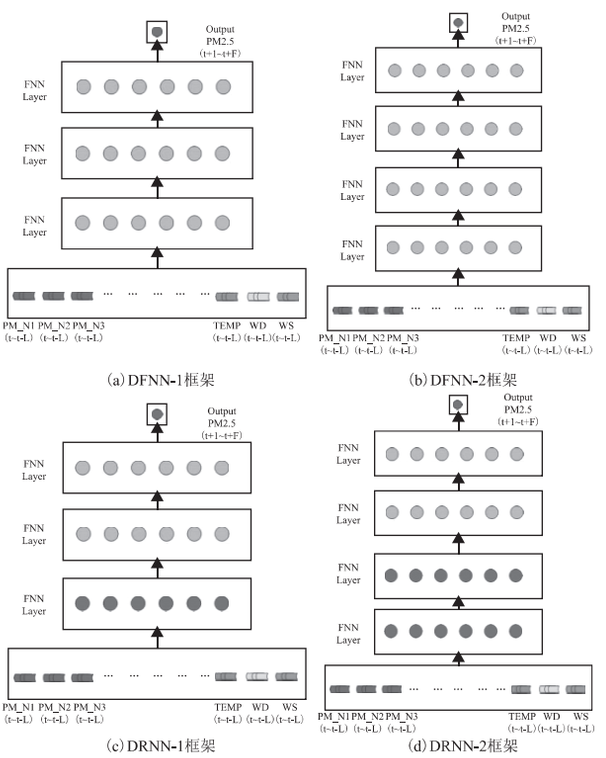

图 3 4 种深度神经网络框架

Fig. 3 4 Deep Neural Network Frameworks

本文使用 Python 和深度学习库 Theano[22] 和 Keras[23]，搭建并训练了上述的深度神经网络结构，总共包含 12 个神经网络，按照数据预处理手段及网络结构可划分为：① 基于前向递补的网络：DRNN1forward，DRNN2forward，DFNN1forward，DFNN2forward；② 基于均值替代的网络：DRNN1mean，DRNN2mean，DFNN1mean，DFNN2mean；③ 基于权重衰减的网络：DRNN1decay，DRNN2decay，DFNN1decay，DFNN2decay，这些网络的结构参数如表 2 所示，

训练参数如表 3 所示。由于深度神经网络含有多个网络层，以及由此带来的大量参数，实际训练中常常存在过拟合的问题，因此本文采用 Dropout 方法来对神经网络进行正则化（regularize）。Dropout 方法由 Srivastava et al. 提出，核心手段是在训练过程中，随机隐去网络中的一些单元以及与之相连的权重连接，从而限制模型单元之间的协同更新[24]。该方法在大量数据集上被验证是有效的。本文使用的 Dropout 比例是 0.1，即含有 Dropout 的网络层在训练过程中，会有 10%的节点（及与之相连的权重连接）会被隐去。

表 2　神经网络结构参数

Tab. 2　Parameters of Neural Networks

缺失值处理手段	基础结构	网络名称	网络层数
前向递补	LSTM、全连接层	DRNN1forward	1 LSTM+2 Dense
		DRNN2forward	2 LSTM+2 Dense
	全连接层	DFNN1forward	3 Dense
		DFNN2forward	4 Dense
均值替代	LSTM、全连接层	DRNN1mean	1 LSTM+2 Dense
		DRNN2mean	2 LSTM+2 Dense
	全连接层	DFNN1mean	3 Dense
		DFNN2mean	4 Dense
权重衰减	LSTM、全连接层	DRNN1decay	1 LSTM+2 Dense
		DRNN2decay	2 LSTM+2 Dense
	全连接层	DFNN1decay	3 Dense
		DFNN2decay	4 Dense

表 3　神经网络模型训练参数

Tab. 3　Parameters of Neural Network Training

训练参数	对应方式与数值
原始数据集大小/条	597 727
记录时间间隔/h	1
训练集大小/(%)	60
验证集大小/(%)	20
测试集大小/(%)	20
预测时长(F，h)	1
历史时长(L，h)	48
最近邻站点数	3
参数优化方式	RMSprop
训练周期数(epoch)	100
batch 大小	256(DRNN)，32(DFNN)
模型损失	均方误差(MSE)

按照对应 DRNN 中 LSTM 层的数量，可以把实现的神经网络划分为两类：① DRNN1 和 DFNN1，对应 DRNN 中只有一层 LSTM；②DRNN2 和 DFNN2，对应 DRNN 中有两层 LSTM。这些模型在测试数据集上的的预测效果见表 4。

表 4　深度神经网络预测精度

Tab. 4　Prediction Accuracy of Deep Neural Networks

模型		RMSE	MAE	IA
DFNN1	DFNN1forward	34.618 8	22.470 1	0.963 037 095 963 954 93
	DFNN1mean	40.398 7	27.129 5	0.942 751 165 479 421 62
	DFNN1decay	37.339 0	24.834 5	0.951 227 925 717 830 66
DRNN1	DRNN1forward	32.374 8	19.046 6	0.966 061 320 155 858 99
	DRNN1mean	37.522 6	23.556 9	0.949 573 792 517 185 21
	DRNN1decay	35.178 6	21.594 4	0.956 022 281 199 693 68
DFNN2	DFNN2forward	32.078 0	19.025 6	0.969 851 322 472 095 49
	DFNN2mean	38.688 7	24.176 4	0.947 031 904 011 964 8
	DFNN2decay	35.467 4	22.100 8	0.958 967 361 599 206 92
DRNN2	DRNN2forward	29.155 7	16.688 3	0.973 376 993 089 914 32
	DRNN2mean	31.177 8	18.751 9	0.967 243 816 703 557 97
	DRNN2decay	29.922 3	17.903 0	0.969 770 222 902 297 97

从表 4 可以看出：① 在 3 种缺失值处理框架中，结合前向递补的深度神经网络框架能达到相对最优的预测精度，结合均值替代的精度相对最低，权重衰减的精度处于二者之间；②在输入数据与网络框架都相同的情况下，DRNN 的精度比 DFNN 更高。以上情况说明：①京津冀地区空气质量的变动情况以突发事件为主，因此在做预测时应更着重考虑短时间内的数据特征；②RNN 能够将时序信息抽象为“状态”进行传递，而在 FNN 中各时刻的数据是等价的，因此前者能更好地把握时间序列的特征。

以结合前向递补的深度神经网络框架为例，4 个模型（DFNN1forward、DFNN2forward、DRNN1forward 和 DRNN2forward）的训练损失分别如图 4(a)～图 4(d)所示，可以看出：在空气污染物时序预测问题上，DFNN 相对 DRNN 而言，训练过程相对更不稳定，在验证集上的预测误差值有较大波动。对于 DFNN 而言，虽然增加神经网络的层数能够在一定程度上提高预测的精度，但是由于对时间序列的特性把握不足，所以对于预测精度的提高能力有限，效果不如 DRNN。

根据历史记录，2014 年 12 月 10 日京津冀地区出现了一次重污染事件，使用 3 种 DRNN1 模型对当日 10 点到 13 点的 PM2.5 区域分布进行预测，得到多站点时间序列预测值再进行距离倒数插值，从而得到时空分布预测结果，见图 5。结合京津冀区域的气象条件分布（温度、湿度、风速、风向）

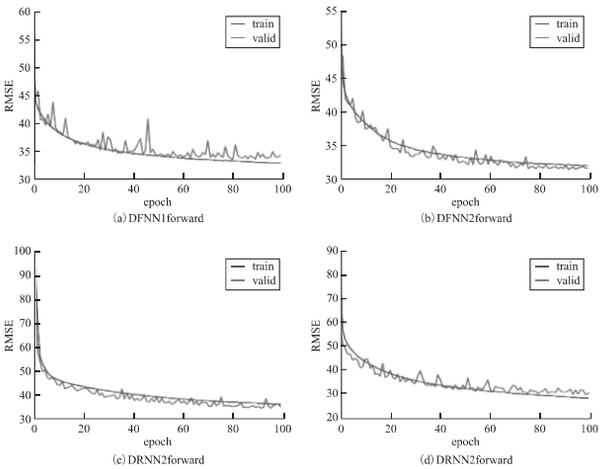

图 4 基于前向递补的深度神经网络训练损失与验证损失值

Fig. 4 Training and Validation Loss of Deep Neural Network Based on Forward-fixing

对时空分布规律进行分析，可以得出结论：①湿度的空间分布与 PM2.5 的空间分布之间存在实时的正相关性，这与雾霾产生时的高湿度条件符合；② 风速的空间分布与 PM2.5 的空间分布之间存在负相关性，且二者的相关变动存在 1～2 h 的时间延迟，表明京津冀区域的雾霾分布受到风速影响很大，而且雾霾在风力影响下的转移需要一定时间；③前向递补能够很好地保持局部突发污染事件的特征，而均值递补则会在一定程度上抹平突发污染的预测结果，权重衰减的结果则介于二者之间，因此对于含有缺失值的空气污染物时间序列进行检测，同时要有效追踪突发重污染情形，前向递补是 3 种缺失值处理算法中的最佳选择。

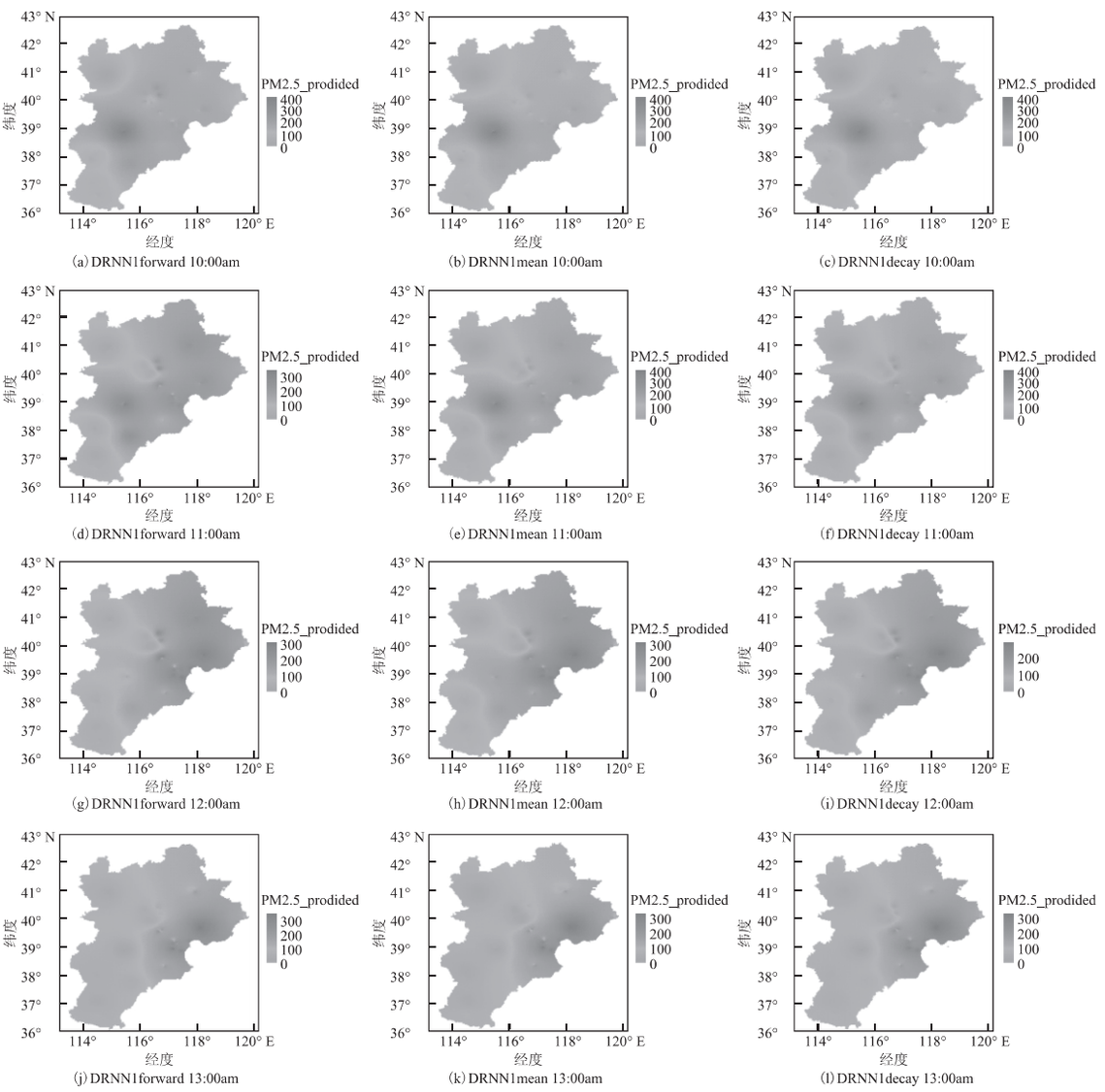

图 5 京津冀单日重污染事件预测结果(2014-12-10 T 10：00am-13：00am)

Fig. 5 Prediction Results Visualization of Air Pollution Event in Jingjinji Area

(2014-12-10 T 10：00am-13：00am)

3 结束语

LSTM 作为一种能够有效处理序列信息的模型，在语音识别、图片自动标注、机器翻译等领域取得了良好的效果。一个有效的空气污染时序预报系统，能够为城市管理决策提供风险预报支持，也可以为城市规划和建设提供理论支持。为了处理时间序列预测中常见的缺失值问题，并提升现有空气污染物时序预报模型的精度，本文结合 LSTM 设计了 3 类能够处理时间序列缺失值的深度神经网络时空预报框架，并在缺失值处理算法的基础上，对基于 LSTM 和基于传统全连接层的深度神经网络的预报效果进行了对比。本文以京津冀地区的全年空气质量及气象数据为样本，训练了 12 个深度神经网络模型，并评估了这些网络在独立测试数据集上的表现，证实了考虑时间序列特性的 DRNN 模型的效果优于不考虑时间信息和系统状态传递的全连接深度神经网络 DFNN，同时也发现基于前向递补的 DRNN 对时间序列有最佳的预报能力。

参考文献

[1] ORGANIZATION W H. Air quality guidelines: global update 2005. Particulate matter, ozone, nitrogen dioxide and sulfur dioxide. [J]. Indian Journal of Medical Research, 2006, 4(4): 492-493.

[2] STIEB D M, BURNETT R T, MARC S D, et al. A new multipollutant, no-threshold air quality health index based on short-term associations observed in daily time-series analyses[J]. Journal of the Air & Waste Management Association, 2008, 58(3): 435-450.

[3] CHEN R, WANG X, MENG X, et al. Communicating air pollution-related health risks to the public: an application of the air quality health index in Shanghai, China [J]. Environment International, 2013, 51(5): 168-173.

[4] BYUN D W, CHING J K S, NOVAK J, et al. Development and Implementation of the EPA's models-3 initial operating version: community multi-scale air quality (CMAQ) model[M]// Air Pollution Modeling and Its Application XII. [S. l.]: Springer US, 1998: 357-368.

[5] GRELL G A, SCHMITZ P R, MCKEEN S A, et al. Fully coupled "online" chemistry within the WRF model[J]. Atmospheric Environment, 2005, 39(37): 6957-6975.

[6] EEFTENS M, BEELEN R, De H K, et al. Development of land use regression models for PM2.5, PM2.5 absorbance, PM10and PM (coarse) in 20 European study areas; results of the ESCAPE project. [J]. Environmental Science & Technology, 2012, 46(20): 11195-11205.

[7] MA Z, HU X, HUANG L, et al. Estimating ground-level PM2.5 in China using satellite remote sensing [J]. Environmental Science & Technology, 2014, 48 (13): 7436-7444.

[8] VOUKANTSIS D, KARATZAS K, KUKKONEN J, et al. Intercomparison of air quality data using principal component analysis, and forecasting of PM 10, and PM 2.5, concentrations using artificial neural networks, in Thessaloniki and Helsinki[J]. Science of the Total Environment, 2011, 409(7): 1266-1276.

[9] OSOWSKI S, GARANTY K. Forecasting of the daily meteorological pollution using wavelets and support vector machine[J]. Engineering Applications of Artificial Intelligence, 2007, 20(6): 745-755.

[10] SÁNCHEZ A B, ORDÓÑEZ C, LASHERAS F S, et al. Forecasting SO_2 pollution incidents by means of elman artificial neural networks and ARIMA models[J]. Abstract & Applied Analysis, 2013(3): 1728-1749.

[11] ONG B T, SUGIURA K, ZETTSU K. Dynamic pre-training of deep recurrent neural networks for predicting environmental monitoring data[C]//IEEE International Conference on Big Data. IEEE, 2014: 760-765.

[12] MA X, YU H, WANG Y, et al. Large-scale transportation network congestion evolution prediction using deep learning theory[J]. 2015, 10(3): 0119044.

[13] CHE Z, PURUSHOTHAM S, CHO K, et al. Recurrent neural networks for multivariate time series with missing values[Z]. [S. l.]: [s. n.], 2016.

[14] KREINDLER D M, LUMSDEN C J. Effects of the irregular sample and missing data in time series analysis [J]. Nonlinear Dynamics Psychology & Life Sciences, 2006, 10(2): 187-214.

[15] WHITE I R, ROYSTON P, WOOD A M. Multiple imputation using chained equations: Issues and guidance for practice[J]. Statistics in Medicine, 2011, 30(4): 377-399.

[16] REHFELD K, MARWAN N, HEITZIG J, et al. Comparison of correlation analysis techniques for irregularly sampled time series[J]. Nonlinear Processes in Geophysics, 2011, 18(3): 389-404.

[17] BENGIO Y, SIMARD P, FRASCONI P. Learning long-term dependencies with gradient descent is difficult[J]. IEEE Transactions on Neural Networks, 1994, 5(2): 157-166.

[18] HOCHREITER S. The vanishing gradient problem during learning recurrent neural nets and problem solutions [J]. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, 1998, 6(2): 107-116.

（下转第 120 页）

5 结束语

基于 Matlab 平台开发了多 GNSS 系统 PPP 解算软件，该软件能够对 GPS、BDS、Galileo 和 QZSS 系统进行单一系统或多系统融合 PPP 解算，并利用 MGEX 数据网的观测数据与产品对开发的软件进行了测试，实验结果与其他机构公布的定位结果精度相当，编制的软件达到了开发要求。

参考文献

[1] 张宝成，欧吉坤，袁运斌，等. 基于 GPS 双频原始观测值的精密单点定位算法及应用[J]. 测绘学报，2010，39(5)：478-483.（ZHANG Baocheng，OU Jikun，YUAN Yunbin. Precise point positioning algorithm based on original dual-frequency gps code and carrier-phase observations and its application[J]. Acta Geodaetica Et Cartographica Sinica，2010，39(5)：478-483.）

[2] 任晓东，张柯柯，李星星，等. Beidou、Galileo、GLONASS、GPS 多系统融合精密单点[J]. 测绘学报，2015，44(12)：1307-1313.（REN Xiaodong，ZHANG Keke，LI Xingxing，et al. Precise point positioning with multi-constellation satellite systems：BeiDou、Galileo、GLONASS、GPS[J]. Acta Geodaetica et Cartographica Sinica，2015，44(12)：1307-1313.）

[3] 戴小蕾，施闯，楼益栋. 多 GNSS 融合精密轨道确定与精度分析[J]. 测绘通报，2016(2)：12-16.（DAI Xiaolei，SHI Chuang，LOU Yidong. Multi-GNSS precise orbit determination and its precision analysis[J]. Bulletin of Surveying and Mapping，2016(2)：12-16.）

[4] LI X，GE M，DAI X，et al. Accuracy and reliability of multi-GNSS real-time precise positioning：GPS，GLONASS，Beidou，and Galileo[J]. Journal of Geodesy，2015，89(6)：607-635.

[5] TEGEDOR J，MELGARD T，VIGEN E，et al. Multi-GNSS precise point positioning including GPS，Glonass，Galileo and BeiDou[C]. PPP：Reaching full potential，2013.

[6] TEGEDOR J，ØVSTEDALO，VIGEN E. Precise orbit determination and point positioning using GPS，Glonass，Galileo and Beidou[J]. Journal of Geodetic Science，2014，4(1)：65-73.

[7] LIU T，YUAN Y，ZHANG B，et al. Multi-GNSS precise point positioning(MGPPP) using raw observations[J]. Journal of Geodesy，2016：1-16.

[8] RABBOU M A，ELRABBANY A. Performance analysis of precise point positioning using multi-constellation GNSS：GPS，GLONASS，Galileo and BeiDou[J]. Survey Review，2016：1-12.

[9] CAI C，GAO Y，PAN L，et al. Precise point positioning with quad-constellations：GPS，BeiDou，GLONASS and Galileo[J]. Advances in Space Research，2015，56(1)：133-143.

[10] 魏子卿. 2000 中国大地坐标系及其与 WGS84 的比较[J]. 大地测量与地球动力学，2008，28(5)：1-5.（WEI Ziqing. China geodetic coordinate system 2000 and its comparison with WGS84[J]. Journal of Geodesy & Geodynamics，2008，28(5)：1-5.）

[11] 罗小敏，蔡昌盛. GPS/GALILEO 组合单点定位精度分析[J]. 大地测量与地球动力学，2013，33(3)：136-140.（LUO Xiaomin，CAI Changsheng. Accuracy assessment of combined GPS/GALILEO single point positioning[J]. Journal of Geodesy & Geodynamics，2013，33(3：136-140).）

[12] 邵佳妮，冯炜，申俊飞. QZSS 系统及其信号设计[J]. 测绘科学，2009(S2)：225-227.（SHAO Jiani，FENG Wei，SHEN Junfei. QZSS and signal design[J]. Science of Surveying & Mapping，2009(S2)：225-227.）

[13] HILLA S，ADAMS G. The GPS Toolbox[J]. GPS Solutions，2000，3(4)：71-74.

（责任编辑：邓国臣）

（上接第 83 页）

[19] HOCHREITER S，SCHMIDHUBER J. Long short-term memory[J]. Neural Computation，1997，9(8)：1735-1780.

[20] GRAVES A. Supervised sequence labelling with recurrent neural networks[M]. Heidelberg：Springer Berlin Heidelberg，2012.

[21] WILLMOTT C J. On the validation of models[J]. Physical Geography，1981，2(55)：184-194.

[22] SRIVASTAVA N，HINTON G，KRIZHEVSKY A，et al. Dropout：a simple way to prevent neural networks from overfitting[J]. Journal of Machine Learning Research，2014，15(1)：1929-1958.

（责任编辑：邓国臣）