计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
Computer Systems & Applications,2022,31(4):221−228 [doi: 10.15888/j.cnki.csa.008493]

E-mail: csa@iscas.ac.cn
http://www.c-s-a.org.cn
Tel: +86-10-62661041

基于深度学习的温度观测数据长时间缺失值插补方法①

郑欣彤[1,2], 边婷婷[3], 张德强[4], 贺　伟[1,2]

[1](中国科学院 地理科学与资源研究所 资源与环境信息系统国家重点实验室, 北京 100101)
[2](中国科学院大学 资源与环境学院, 北京 100049)
[3](北京联合大学 管理学院, 北京 100101)
[4](中国科学院 华南植物园鼎湖山森林生态系统定位研究站, 广州 510650)
通信作者: 边婷婷, E-mail: teacherbian@126.com

摘　要: 完整高精度的温度观测数据是农业气象灾害监测、生态系统模拟重要的输入参数. 由于野外气象观测条件的限制, 气象观测数据缺失现象是常态, 数据插补方法是气象数据应用必要处理步骤. 本文针对野外小气象观测站站点半小时温度观测数据长时间缺失值问题, 结合同一地点较低频次的人工温度观测, 构建了新的温度缺失值插补深度学习模型, 对缺失的半小时温度观测数据进行高精度插补. 本文构建的深度学习模型, 采用了基于编码-解码结构的序列-序列深度学习结构 (BiLSTM-I), 模型编码层采用双向 LSTM-I 网络, 解码层分别采用 LSTM 解码结构与全连接两种解码结构. 试验分析结果表明, 本文设计的 BiLSTM-I 深度学习温度插补方法要优于其他方法, 可满足了高精度温度数据插补需要, 而且 LSTM 解码结构的 BiLSTM-I 模型具有更好的数据插补精度. 文章最后还分析了 BiLSTM-I 深度学习模型的泛化能力, 结果表明 BiLSTM-I 模型具有不同温度缺失窗口长度的插补能力.

关键词: 长时间序列; BiLSTM-I; 温度缺失; 高精度插补; 深度学习; 长记忆

引用格式: 郑欣彤,边婷婷,张德强,贺伟.基于深度学习的温度观测数据长时间缺失值插补方法.计算机系统应用,2022,31(4):221–228. http://www.c-s-a.org.cn/1003-3254/8493.html

Interpolation of Long Time Missing Values of Temperature Based on Deep Learning

ZHENG Xin-Tong[1,2], BIAN Ting-Ting[3], ZHANG De-Qiang[4], HE Wei[1,2]

[1](State Key Laboratory of Resource and Environmental Information System, Institute of Geographic Sciences and Natural Resources Research, Chinese Academy of Sciences, Beijing 100101, China)
[2](College of Resources and Environmental, University of Chinese Academy of Sciences, Beijing 100049, China)
[3](Management College, Beijing Union University, Beijing 100101, China)
[4](DinghuShan Forest Ecosystem Research Station, South China Botanical Garden, Chinese Academy of Science, Guangzhou 510650, China)

Abstract: Complete and high-precision temperature observation data are important input parameters for agro-meteorological disaster monitoring and ecosystem simulation. Due to the limitation of meteorological field observation conditions, missing meteorological observation data is common. In response, interpolation becomes a necessary processing step before meteorological data application. In this study, we construct a new deep learning model for interpolation of missing temperature data, which is employed to interpolate the missing half-hour temperature observations with high accuracy together with the low-frequency manual temperature observations at the same location. The deep learning model has a sequence-to-sequence deep learning structure based on the coding-decoding structure. A bidirectional LSTM-I (BiLSTM-I) network is used for the coding layer of the model, and an LSTM decoding structure and a fully connected decoding structure are respectively adopted for the decoding layer. The experimental analysis

① 基金项目: 国家重点研发计划 (2107YFD0300403)
收稿时间: 2021-07-04; 修改时间: 2021-07-30, 2021-08-12; 采用时间: 2021-09-29; csa 在线出版时间: 2022-03-22

results show that the designed BiLSTM-I deep learning method for temperature interpolation is better than other methods. It can meet the need for high-precision temperature data interpolation. Particularly, the BiLSTM-I model with the LSTM decoding structure has higher data interpolation precision. The generalization ability of the BiLSTM-I deep learning model is also explored, and the results show that the model is effective in data interpolation for different lengths of the temperature missing window.

Key words: long time series; BiLSTM-I; temperature missing; high-precision interpolation; deep learning; long term memory

1 前言

温度是农业、生态系统研究非常重要的观测量, 农业作物生长的模拟、农业气象灾害监测、生态系统模拟中温度是必不可少的输入[1,2]. 随着农业、生态模拟的精细化, 要求温度数据具有更高的精度, 如农业气象灾害干热风监测、林块生态系统碳排放的模拟等[3,4], 高精度的温度观测量是必不可少的输入参数. 温度观测数据一般通过野外气象观测站获取, 由于设备故障、恶劣环境或是认为操作失误等原因, 小气象观测难免会出现缺失[5], 缺失数据插补或补全, 是温度观测数据运用前必不可少的预处理工作.

本文针对中国广州一个森林生态站长时间间隔温度观测数据缺失进行插补方法研究. 由于该森林生态站处在雷雨区, 小气象站夏季容易因恶劣天气损坏, 容易造成较长时间的数据缺失. 论文选择了该生态站同时有自动观测气象站, 作为比对观测, 该小气象场还有人工温度观测设施. 自动观测气象数据输出的观测产品时间频率为 30 分钟, 每天有 48 条观测记录数据; 人工观测分早、中、晚每天 3 次, 产生 3 条记录. 本文研究的实际应用问题: 如何运用不同数据插补方法, 通过每天低频的人工温度观测数据, 获取完整的高精度半小时频率温度观测数据.

数据插补是众多学科数据分析前必不可少的预处理工作. 目前以数据插值、统计分析和时间序列分析等为基础, 发展出了多种数据插补方法[6,7], 但对高精度数据插补研究还处初步阶段[5]. 高精度数据插补的要求是从已观测数据中学习到数据的规律或缺值模式, 从而实现对未观测数据的准确估计. 深度学习是机器学习领域一个新的研究方向, 是人工智能领域的一项颠覆性技术创新, 除了带来图像、语音和自然语言处理领域的突破, 也成功应用到了众多学科领域[8,9]. 深度学习旨在获得样本数据的内在规律和表示[10], 和数据插补的需求非常契合.

深度学习技术已经在交通、医疗、传感器网络等多个领域的数据插补中获得了成功应用[11−13], 并发展出了 GRU、LSTM、GAN 等不同结构的数据插补深度学习神经网络[14]. GRU (gate recurrent unit) 和 LSTM (long short-term memory) 都是循环神经网络的不同形式, 可以解决 RNN 网络学习过程中的梯度消失或爆炸问题[15]. 这两种结构神经网络在数据插补应用中, 不但可以从已观测数据之中学习规律, 也可从数据缺失值模式中进一步学习, 提高数据插补精度[16,17]. GAN (generative adversarial networks) 网络用于学习多变量时间序列的总体分布, 从而对观测数据中的缺失值进行插补[18].

在众多深度学习时间序列数据插补模型中, 一种基于序列-序列 (Seq2Seq) 的深度学习模型在多个标准样本集数据插补都有很好的表现[19], 该结构采用了双向循环 LSTM 网络, 在随后的实际应用也进一步验证了该结构适用于时间序列数据缺失插补问题[20]. 另外, 基于 Encoder-Decoder 结构的深度学习神经网络在数据插补方面也获得成功应用[21]. 这些不同结构的深度学习模型为本文研究提供了重要参考.

本文运用低频人工温度观测数据, 来插补高频次机器观测数据的长时间观测值缺失问题. 时间序列数据缺失值插补虽然已很丰富, 但针对这一特定应用场景的数据插补方法研究文献还是较少[22]. 下面是一个具体的生态台站的小气象观测数据, 该小气象站同时具有半小时自动温度观测和每天 3 次人工温度观测, 但半小时自动温度观测数据存在较长时间的缺失值. 为了实现半小时温度观测数据的高精度插补, 文中详细给出了一个编码-解码结构的序列-序列深度学习温度插补模型的构建过程和数学公式, 并将其与其他插补方法进行了插补精度对比分析.

2 研究数据介绍

本文研究采用了我国广州鼎湖山森林生态系统国家野外科学观测研究站的气象温度观测数据. 该生态系统观测站开展有温度观测对比试验, 同时开展人工观测和气象机器自动观测活动, 有较长时间的温度观测数据记录, 表 1 是用于本文研究的温度人工观测数据和自动机器观测数据情况.

表 1 温度观测数据集信息表

数据集	观测频率(每天)	时间范围	缺失值情况
温度人工观测数据1	早8点	2018/11/13–2020/2/10	无
温度人工观测数据2	下午2点	2018/11/13–2020/2/10	无
温度人工观测数据3	晚上8点	2018/11/13–2020/2/10	无
机器自动气象观测数据4	每半小时	2018/11/13–2020/2/10	既有短时间温度缺失, 有一次长时间温度缺失

由于鼎湖山生态站位于中国南方的山区, 自动观测设备容易受雷雨季节影响而产生较长时间观测记录的缺失. 图 1 是某一机器自动气象观测数据的数据缺失情况分布图, 从图中可见在 2020 年 7 月有一次超过 2 个月的温度观测数据缺失.

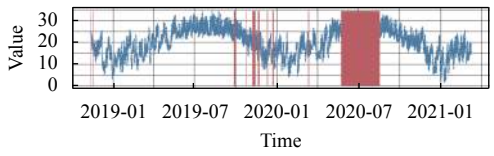

图 1 半小时温度观测数据缺失值分布图

运用每天 3 次的人工观测数据对 2020 年 7 月超过 2 个月的机器温度观测数据缺失进行插补, 是本文方法研究的应用背景. 由于人工观测数据和机器观测数据之间很容易建立线性相关关系, 本文数据插补问题研究核心是如何运用低频的人工温度观测数据对高频的机器自动温度观测数据进行插补.

3 温度观测数据深度学习插补

3.1 基本定义

气温具有天的周期性, 很自然的将长时间序列温度观测数据按照天为单位进行划分, 变为每天 48 个观测值的分段序列. 为了更好的把研究集中到长时间间隔缺失值的插补, 对时间序列中偶尔或短时间数据缺先采用上文的卡尔曼平滑的方法进行插补. 这样以天为分段的温度时间序列包含两种, 即没有缺失值的每天分段, 记为d^j_{full}, 和只包含早中晚 3 次观测值的每天分段, 记为d^j_{miss}, 这样以天为分段的时间分段序列可以表示为:

$$\{d^1_{\text{full}},\cdots,d^i_{\text{full}},d^{i+1}_{\text{miss}},\cdots,d^{i+m}_{\text{miss}},d^{i+m+1}_{\text{full}},\cdots,d^n_{\text{miss}}\} \quad (1)$$

序列 (1) 表示长度为 n 天温度时间序列中, 缺失值窗口宽度为 m 天. 半小时温度观测序列 (1) 为长度为 $48n$, 存在缺失值 $48m$ 的半小时温度观测数据序列, 以天分段后的表达形式.

为了表示时间序列 (1) 中缺失值的位置, 对长度为 $L(48n)$ 的半小时采样温度时间序列$\{T^i_t\}$, 构建相应长度为 L 的掩码时间序列$\{m^i_t\}$, 其中:

$$m^i_t=\begin{cases}0, \text{如} T^i_t \text{未被观测}\\ 1, \text{否则}\end{cases}$$

现在以天为单位, 对长度为 L 的半小时掩码序列进行分段, 没有缺失值的掩码每天分段记为M^j_{full}, 和只包含早中晚 3 次观测值的掩码每天分段, 记为M^j_{miss}, 这样就可以建立与式 (1) 对应的以天为分段的掩码序列:

$$\{M^1_{\text{full}},\cdots,M^i_{\text{full}},M^{i+1}_{\text{miss}},\cdots,M^{i+m}_{\text{miss}},M^{i+m+}_{\text{full}},\cdots,M^n_{\text{full}}\} \quad (2)$$

3.2 滚动窗口采样

采用滚动窗的方法, 基于以天为分段的时间序列为深度学习模型训练构建样本集. 对长度为 m (天) 的缺失值进行插补, 需构建样本滚动窗口的长度大于 m, 并且在 m 的两端各保留长度为 s (天) 的观测数据, 这样滚动窗口长度 w 为 m+2×s 天. 训练样本为适应序列-序列 (Seq2Seq) 的训练方法来构建, 对长度为 w 的训练输入样本中温度观测序列为:

$$\{d^1_{\text{full}},\cdots,d^s_{\text{full}},d^{s+1}_{\text{miss}},\cdots,d^{s+m}_{\text{miss}},d^{s+m+}_{\text{full}},\cdots,d^w_{\text{miss}}\} \quad (3)$$

可通过训练形成如下的时间序列结果输出:

$$\{d^1_{\text{full}},\cdots,\hat{d}^s_{\text{full}},\hat{d}^{s+1}_{\text{impt}},\cdots,\hat{d}^{s+m}_{\text{impt}},d^{s+m+}_{\text{full}},\cdots,d^w_{\text{miss}}\} \quad (4)$$

序列 (4) 中, $\hat{d}^j_{\text{impt}}$为对缺失值插补后, 某一天每半小时温度观测数据值的完整分段. 在构建训练样本时, 按照观测值与掩码序列以天为顺序对应关系, 构建长度为 w (天) 的掩码序列, 作为训练样本的另一个输入.

训练样本需要在没有缺失值的温度观测序列基础上构建, 样本中观测值缺失的模式同实际情况一致, 即每天只有早中晚 3 次观测值. 表 2 是训练样本中存在缺失值的某一天温度数据及其对应的掩码示例.

表 2 样本序列中缺失值窗口内某一天的数据示例

1	2	…	17	18	…	27	28	29	…	39	40	41	…	47	48
Na	Na	…	17.14	Na	…	Na	21.78	Na	…	Na	19.86	Na	…	Na	Na
0	0	…	1	0	…	0	1	0	…	0	1	0	…	0	0

注: 表中第1行是以半小时为单位的时间序号, 从0点开始; 第2行为温度值, 单位℃, 只有早、中、晚3次有效观测值; 第3行为温度观测值缺失位置掩码.

3.3 深度学习模型的设计

典型的基于 Seq2Seq 的时间序列数据插补深度学习模型有 SSIM, BRTS-I 等[19,21]. 本文吸收了这些模型的优点, 将 Seq2Seq 和 Encoder-Decoder 深度学习架构结合起来, 所设计的深度学习模型结构如下文所述. 上面输入序列 (1) 被记为 $x=\{x_1, x_2, \cdots, x_n\}$, 输出序列 (4) 被记为 $y=\{y_1, y_2, \cdots, y_n\}$, 掩码序列 (2) 被记为 $m=\{m_1, m_2, \cdots, m_n\}$.

(1) 编码

从图 2 中可见, 深度学习结构中的编码部分的基本结构是 LSTM-I, 该结构与 BRTS-I 结构中的 RTS-I 结构相似, 其中的循环神经网络单元直接采用了长短期记忆单元; 另外, 本文温度观测缺失值部分, 每天 48 个半小时温度值, 只有 3 个观测值, 有效值比较稀疏, 所有没有采用 RTS-I 中的缺失值时间间隔的变量和相应的训练公式. 下面定义中 LSTM 被简化为一个简单算子的形式, 将 LSTM-I 单元过程数学描述为:

$$\tilde{x}_t = W_x h_{t-1} + b_x \tag{5}$$

$$x_t^c = x_t \odot m_t + (1 - m_t) \odot \tilde{x}_t \tag{6}$$

$$h_t = LSTM(x_t^c, h_{t-1}) \tag{7}$$

$$l_t = \langle m_t, \pounds(x_t, \tilde{x}_t) \rangle \tag{8}$$

式 (5) 将上一个 LSTM 单元的隐状态 h_{t-1} 转化为估计向量 $\tilde{x}_t$, 其中 W_x、b_x 为模型参数; 式 (6) 通过运用掩码向量 m_t, 把输入向量 x_t 中的缺失值替换为估计向量 $\tilde{x}_t$ 对应的值; 式 (7) 通过 LSTM 网络单元把 x_t^c 和隐状态 h_{t-1} 产生预测状态 h_t; 式 (8) 是 LSTM-I 单元的估计误差, 为缺失值位置上观测值与估计值绝对差的累计量.

图 2 中神经网络编码部分由双向的 LSTM-I 神经网络构成: 一个是从时间序列的开始到结束读取输入, 产生前向隐状态向量序列 $\overrightarrow{h} = \{\overrightarrow{h}_1, \overrightarrow{h}_2, \cdots, \overrightarrow{h}_n\}$; 另一个是从时间序列的结束到开始反向读取输入, 产生后向隐状态序列 $\overleftarrow{h} = \{\overleftarrow{h}_1, \overleftarrow{h}_2, \cdots, \overleftarrow{h}_n\}$. 将前向和后向隐状态序列拼接到一起, 构成编码层的编码输出 $h=\{h_1, h_2, \cdots, h_n\}$, 其中向量 h_i 为:

$$h_i = \{\overrightarrow{h}_i, \overleftarrow{h}_i\} \tag{9}$$

双向编码 LSTM-I 编码网络误差包括正向和逆向估计误差两部分.

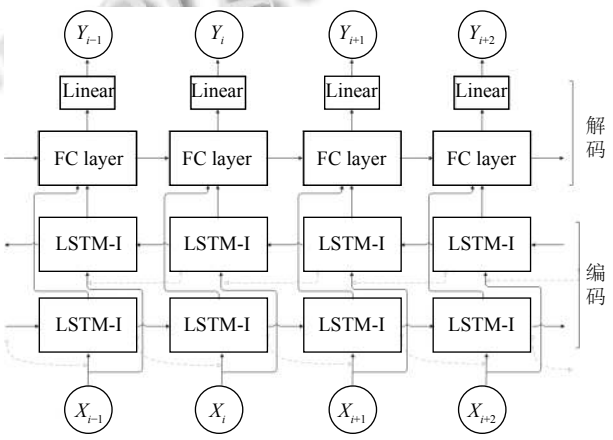

图 2 温度值缺失值插补神经网络结构图

(2) 解码

解码层接收编码输出序列 h, 产生插补后的时间序列结果序列 y. 本文通过优选, 采用了 2 种解码结构: 一种是全连接层和一层线性层组合, 如图 3 所示; 另一种是 LSTM 和一层线性层的组合, 如图 4 所示.

① 基于两层线性层组合解码过程数学描述如下:

$$\tilde{h}_t = Dropout(h_t) \tag{10}$$

$$s_t = g(W_s \tilde{h}_t + b_s) \tag{11}$$

$$y_t = W_y s_t + b_y \tag{12}$$

$$l_y = \langle m_t, \pounds(x_t, y_t) \rangle \tag{13}$$

式 (10) 中的 *Dropout* 通过对解码输入随机丢弃部分神经元, 能够起到预防过拟合的作用; 式 (11) 为全连接层, g为激活函数, 全连接层产生输出状态序列 $s=\{s_1, s_2, \cdots, s_n\}$; 由于温度值是连续值, 式 (12) 为最上层即线性变换层, 输出插补结果序列 y; 式 (13)是解码的插补结果误差, 为缺失值位置上观测值与插补值绝对差的累计量结果.

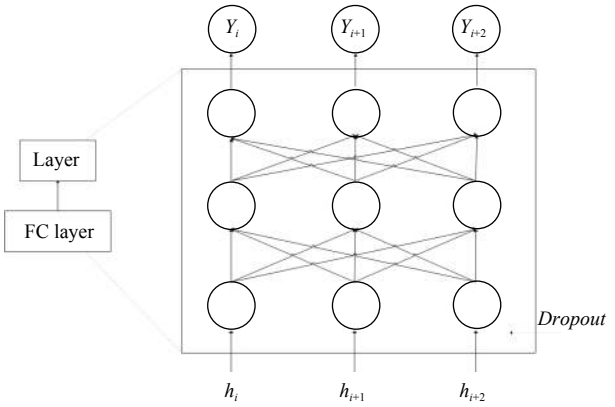

图 3 全连接层和线性层组合的解码详细结构

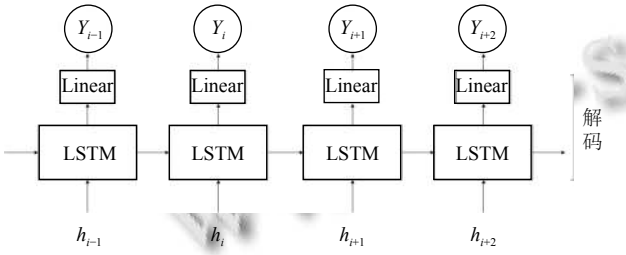

图 4 LSTM 层和线性层组合的解码结构

② LSTM 层和线性层组合的解码结构

LSTM 层和线性层组合的解码结构如图 4 所示. 解码层接收编码层的输出序列 h, 产生插补后的时间序列结果序列 y.

LSTM 层和线性层组合的解码过程数学描述为:

$$s_t = LSTM(h_t, s_{t-1}) \tag{14}$$

$$y_t = W_y s_t + b_y \tag{15}$$

$$l_y = \langle m_t, £(x_t, y_t)\rangle \tag{16}$$

如式 (14), 解码层底部是一个标准的 LSTM 网络, 该网络综合编码输出序列 h, 产生包含更丰富信息输出状态序列 $s=\{s_1, s_2, \cdots, s_n\}$; 如式 (15), 由于温度值是连续值, 解码层顶部采用了线性全连接层, 输出插补结果序列 y. 同式 (13), 式 (16) 是解码层的插补结果误差.

上述两种解码机制的数据插补深度学习网络的误差构成是相同的, 神经网络的误差包括 3 部分, 即:

$$l_t = l_t^f + l_t^b + l_y \tag{17}$$

式 (17) 中, l_t^f 前向 LSTM-I 编码层的估计误差, l_t^b 为后向 LSTM-I 编码层的估计误差.

4 插值效果评估方法

本文采用多个指标评价不同数据插补方法的性能, 评价指标的数值在测试样本集上计算. 包括均方根误差 (*RMSE*), 平均绝对误差 (*MAE*), 平均相对误差 (*MRE*) 和皮尔逊相关系数 (*PCC*), 定义如下:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - y_i)^2} \tag{18}$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|x_i - y_i| \tag{19}$$

$$MRE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{x_i - y_i}{x_i}\right| \tag{20}$$

$$PCC = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2 \sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{21}$$

上面指标计算中, x_i 为所构造测试本中缺失值位置的实际观测值, y_i 为缺失值位置对应的插补结果值. *PCC* 评价指标计算如式 (21), 其中$\bar{x}$为样本中缺失值位置实际观测值的总体平均, $\bar{y}$为缺失值位置插补结果的总体平均.

5 方法结果

作为对比, 本文选用了简单的总体平均插补方法 (Mean); 基于时间序列分解的卡尔曼插补方法 (Kalman-struct); 基于深度学习的 BRTS-I 时间序列插补方法; 本文设计的两种编码-解码结构的双向 LSTM 网络插补方法, 解码层为全连接 (BiLSTM-FC-I) 和解码层为 LSTM (BiLSTM-LSTM-I).

上述方法中, 总体平均插补方法可以在整个数据集上计算获取, 而插补方法 BRTS-I、BiLSTM-FC-I、BiLSTM-LSTM-I、Kalman-struct 均需要先把整个数据集分为训练集和测试集, 然后在相同的训练集上进行训练, 在同一测试集上进行精度分析. 基于深度学习的 BRTS-I、BiLSTM-FC-I、BiLSTM-LSTM-I 虽然模型结构有所不同, 但其关键参数 LSTM 状态的维度均相同, 均取值为 108; 另外模型的训练参数也相同, 采用了相同的 mini batch 参数和优化方法, 优化方法均选用 Adam, 初始学习率取值为 0.001; 训练终止策略均为连续 10 轮测试精度均不超过训练过程中测试精度的最优值. 深度模型实现是以 PyTorch 深度学习框架为基础, 以 2020 年 07 月缺失值窗口左侧的观测数据构建

训练集, 右侧的观测数据构建测试集. 深度学习插补方法构建了两种训练样本, 一种缺失值时间窗口设定为 30 天, 另一种缺失值时间窗口设定为 60 天, 这两种训练样本缺失值窗口前后连续观测值均设定为 14 天. 为了区别这两种训练样本, 下面将缺失值时间窗口长度作为了深度学习插补方法的后缀, 将各种插补方法的结果总结如表 3.

表 3 时间序列插补方法结果统计表

方法名称	*RMSE* (℃)	*MAE* (℃)	*MRE*	*PCC*
BiLSTM-LSTM-I-60	0.4929	0.3319	0.0173	0.9963
BiLSTM-LSTM-I-30	0.4686	0.3215	0.0170	0.9968
BiLSTM-FC-I-60	0.7032	0.5135	0.0268	0.9925
BiLSTM-FC-I-30	0.7649	0.5737	0.0303	0.9917
BRTS-I-60	1.3959	1.0300	0.0537	0.9697
BRTS-I-30	1.3724	1.0177	0.0538	0.9726
Kalman-Struct	1.1742	0.8449	0.0472	0.9873
Mean	6.5614	5.2398	0.2927	—

从表 3 中可见, 深度学习方法要明显优于简单的总体平均方法. 深度学习方法之间的精度也存在较大的差别, 图 5 是卡尔曼插值方法, 以及各种深度学习方法插值 *RMSE* 精度的比较图.

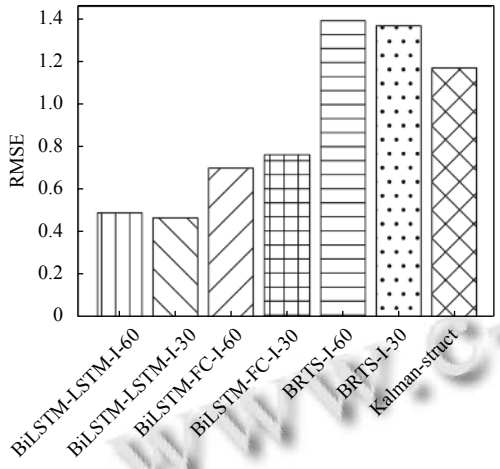

图 5 不同插值方法 *RMSE* 精度的对比图

从图 5 中比较 BRTS-I、卡尔曼方法、BiLSTM-I 三种方法, 本文设计的两种 BiLSTM-I 深度学习气温插补方法要优于其他两种方法; 基于 LSTM 解码的 BiLSTM-I 深度学习方法精度优于基于全连接解码的 BiLSTM-I 深度学习方法; BRTS-I 深度学习时间序列插补方法精度最低.

卡尔曼平滑方法时间序列插补方法精度取决于状态方程是否准确表达了时间序列, Kalman-struct 假定时间序列的趋势性和季节性的成分可通过基本的线性方程进行拟合. 深度学习方法, 没有对时间序列的表达形式进行任何假设, 而是通过训练数据集, 自动学习时间序列的准确表达形式. 从测试结果看, BiLSTM-I 方法比 Kalman-struct 卡尔曼方法, 以及 BRTS-I 深度学习方法更有可能获取了时间序列的准确表达, 从而获得了更高的数据插补精度.

本文设计的两种解码结构的 BiLSTM-I 模型插补精度都高于 BRTS-I 模型. BiLSTM-I 模型与 BRTS-I 模型的区别主要有两点: 首先从模型结构上, BiLSTM-I 采用了 Encoder-Decoder 结构, 而 BRTS-I 只相当于 BiLSTM-I 模型的 Encoder 部分, 模型结构上 BiLSTM-I 有利于充分学习数据的潜在分布规律, 从而可以提高数据插补精度; 其次模型代价函数有区别, BiLSTM-I 和 BRTS-I 的代价函数均由 3 部分构成[19], 前两部分是相同的, BRTS-I 模型代价函数的第 3 部分是前向和后向 LSTM 网络对缺失估计值的差; BiLSTM-I 模型代价函数的第 3 部分则为最后的估计值和真实观测值的差, BiLSTM-I 模型误差函数对插补结果的评价更直接, 模型收敛误差和插补精度直接对应, 从而确保模型收敛时插补误差也能达到最小.

两种解码结构的 BiLSTM-I 模型, 基于 LSTM 解码的模型插补精度优于全连接解码结构模型插补精度, 这主要是 LSTM 解码时, 不但可以利用当前的编码输出信息, 还可以利用之前的编码输出信息; 而全连接编码则只能利用当前的编码信息, LSTM 解码方法对编码信息的利用更为充分.

图 5 中两种解码结构的 BiLSTM-I 模型, 缺值窗口分别为 30 天和 60 天的测试精度基本一致. 深度学习方法应用中模型的泛化能力非常重要, 本文问题体现在模型对不同宽度缺失值窗口插补精度是否一致. 为了对这一点进行检验, 我们用缺失值窗口为 30 天的模型对缺失值为 60 天时间温度观测时间序列进行插补, 然后用缺失值窗口为 60 天的模型对缺失值为 30 天时间温度观测时间序列进行插补, 表 4 是这两种情况下, 两种解码结构的插补方法结果的精度统计表.

从表 4 可见, 无论是缺失值窗口为 60 天的模型应用到缺失值情况为 30 天, 还是缺失值窗口为 30 天模型应用到缺失值情况为 60 天, 两种解码结构的模型精度的各项指标都非常稳定, 这表明这两种解码结构的 BiLSTM-I 深度学习模型都对不同缺失值窗口有较好的泛化能力.

表 4 BiLSTM-I 模型分别应用到缺失值为 30、60 天的插值精度统计表

缺失值时间窗口	插补模型	*RMSE*	*MAE*	*MRE*	*PCC*
30天	BiLSTM-LSTM-I-30	0.4686	0.3215	0.0170	0.9968
	BiLSTM-LSTM-I-60	0.4865	0.3326	0.0176	0.9966
	BiLSTM-FC-I-30	0.7649	0.5737	0.0303	0.9917
	BiLSTM-FC-I-60	0.6864	0.5032	0.0266	0.9933
60天	BiLSTM-LSTM-I-60	0.4929	0.3319	0.0173	0.9963
	BiLSTM-LSTM-I-30	0.4834	0.3293	0.0172	0.9964
	BiLSTM-FC-I-60	0.7032	0.5135	0.0268	0.9925
	BiLSTM-FC-I-30	0.7899	0.5914	0.0308	0.9907

6 总结

本文运用不同深度学习数据插补方法, 通过每天低频的人工温度观测数据, 获取完整的高精度半小时频率温度观测数据. 本文采用序列-序列的时间序列插补方法, 基于编码-解码结构的深度学习模型 (BiLSTM-I), 编码层采用双向 LSTM-I 网络, 解码层分别采用 LSTM 解码结构与全连接解码结构, 设计了两种解码结构的深度学习数据插补模型.

试验分析结果表明, 本文设计的 BiLSTM-I 深度学习气温插补方法要优于其他方法. 基于 LSTM 解码结构的深度学习模型, 缺失值时间为 30 天的测试集, 测试结果精度 *RMSE* 为 0.47℃; 缺失值时间为 60 天的测试集, 测试结果精度 *RMSE* 为 0.49℃. 基于全连接解码结构的深度学习模型, 缺失值时间为 30 天的测试集, 测试结果精度 *RMSE* 为 0.76℃; 缺失值时间为 60 天的测试集, 测试结果精度 *RMSE* 为 0.70℃.

最后, 文章还分析了 BiLSTM-I 深度学习插补方法对不同时间温度缺失长度的适应能力. 分别用缺失值时间长度为 30 天的训练模型, 对缺失值为 60 天的测试集进行插补; 以缺失值时间长度为 60 天的训练模型, 对缺失值为 30 天的测试集进行插补, 结果表明两种解码结构的深度学习训练模型对不同的温度缺失时间长度具有泛化能力.

参考文献

1 Lara-Estrada L, Rasche L, Sucar LE, *et al*. Inferring missing climate data for agricultural planning using Bayesian networks. Land, 2018, 7(1): 4. [doi: 10.3390/land7010004]

2 Huang MT, Piao SL, Ciais P, *et al*. Air temperature optima of vegetation productivity across global biomes. Nature Ecology & Evolution, 2019, 3(5): 772–779. [doi: 10.1038/s41559-019-0838-x]

3 Hu LW, He HL, Shen Y, *et al*. Modeling the carbon cycle of a subtropical Chinese fir plantation using a multi-source data fusion approach. Forests, 2020, 11(4): 369. [doi: 10.3390/f11040369]

4 Luedeling E. Interpolating hourly temperatures for computing agroclimatic metrics. International Journal of Biometeorology, 2018, 62(10): 1799–1807. [doi: 10.1007/s00484-018-1582-7]

5 Afrifa-Yamoah E, Mueller UA, Taylor SM, *et al*. Missing data imputation of high-resolution temporal climate time series data. Meteorological Applications, 2020, 27(1): e1873. [doi: 10.1002/met.1873]

6 Beck MW, Bokde N, Asencio-Cortés G, *et al*. R package imputeTestbench to compare imputation methods for univariate time series. The R Journal, 2018, 10(1): 218–233. [doi: 10.32614/RJ-2018-024]

7 Lepot M, Aubin JB, Clemens FHLR. Interpolation in time series: An introductive overview of existing methods, their performance criteria and uncertainty assessment. Water, 2017, 9(10): 796. [doi: 10.3390/w9100796]

8 Hatcher WG, Yu W. A survey of deep learning: Platforms, applications and emerging research trends. IEEE Access, 2018, 6: 24411–24432. [doi: 10.1109/ACCESS.2018.2830661]

9 Pouyanfar S, Sadiq S, Yan YL, *et al*. A survey on deep learning: Algorithms, techniques, and applications. ACM Computing Surveys, 2019, 51(5): 92. [doi: 10.1145/3234150]

10 LeCun Y, Bengio Y, Hinton G. Deep learning. Nature, 2015, 521(7553): 436–444. [doi: 10.1038/nature14539]

11 Duan YJ, Lv YS, Kang WW, *et al*. A deep learning based approach for traffic data imputation. 17th International IEEE Conference on Intelligent Transportation Systems (ITSC). Qingdao: IEEE, 2014. 912–917. [doi: 10.1109/ITSC.2014.6957805]

12 Gad I, Hosahalli D, Manjunatha B R, *et al*. A robust deep learning model for missing value imputation in big NCDC dataset. Iran Journal of Computer Science, 2021, 4(2): 67–84. [doi: 10.1007/s42044-020-00065-z]

13 Matusowsky M, Ramotsoela DT, Abu-Mahfouz AM. Data imputation in wireless sensor networks using a machine learning-based virtual sensor. Journal of Sensor and Actuator Networks, 2020, 9(2): 25. [doi: 10.3390/jsan9020025]

14 Muhammad S. Deep learning based approaches for imputation of time series models [Master's thesis]. Waterloo: University of Waterloo, 2020.

15 Hochreiter S, Schmidhuber J. Long short-term memory.

Neural Computation, 1997, 9(8): 1735–1780. [doi: 10.1162/neco.1997.9.8.1735]

16 Che ZP, Purushotham S, Cho K, *et al*. Recurrent neural networks for multivariate time series with missing values. Scientific Reports, 2018, 8(1): 6085. [doi: 10.1038/s41598-018-24271-9]

17 Song W, Gao C, Zhao Y, *et al.* A time series data filling method based on LSTM—Taking the stem moisture as an example. Sensors, 2020, 20(18): 5045. [doi: 10.3390/s20185045]

18 Luo YH, Zhang Y, Cai XR, *et al*. E^2GAN: End-to-end generative adversarial network for multivariate time series imputation. Proceedings of the 28th International Joint Conference on Artificial Intelligence. Macao: IJCAI, 2019. 3094–3100. [doi: 10.24963/ijcai.2019/429]

19 Cao W, Wang D, Li J, *et al*. BRITS: Bidirectional recurrent imputation for time series. Proceedings of the 32nd International Conference on Neural Information Processing Systems. Montreal: ACM, 2018. 6776–6786. [doi: 10.5555/3327757.3327783]

20 Dabrowski JJ, Rahman A. Sequence-to-sequence imputation of missing sensor data. 32nd Australasian Joint Conference on AI 2019: Advances in Artificial Intelligence. Adelaide: Springer, 2019. 265–276. [doi: 10.1007/978-3-030-35288-2_22]

21 Zhang YF, Thorburn PJ, Xiang W, *et al*. SSIM—A deep learning approach for recovering missing time series sensor data. IEEE Internet of Things Journal, 2019, 6(4): 6618–6628. [doi: 10.1109/JIOT.2019.2909038]

22 Li ZN, Yu H, Zhang GH, *et al*. A Bayesian vector autoregression-based data analytics approach to enable irregularly-spaced mixed-frequency traffic collision data imputation with missing values. Transportation Research Part C:Emerging Technologies, 2019, 108: 302–319. [doi: 10.1016/j.trc.2019.09.013]