计 算 机 研 究 与 发 展
Journal of Computer Research and Development
https://crad.ict.ac.cn
62(9): 2233−2258, 2025

基于深度学习的不完整时序数据补全方法综述

周　立[1]　吴洋洋[2]　苗晓晔[1]

[1]（浙江大学数据科学研究中心　杭州　310058）

[2]（浙江大学软件学院　浙江宁波　315100）

（zhou_li@zju.edu.cn）

Survey of Incomplete Time Series Imputation Methods Based on Deep Learning

Zhou Li[1], Wu Yangyang[2], and Miao Xiaoye[1]

[1]（*Center for Data Science*, *Zhejiang University*, *Hangzhou* 310058）

[2]（*School of Software Technology*, *Zhejiang University*, *Ningbo*, *Zhejiang* 315100）

Abstract　Time series are widely present in domains such as industry, finance, transportation, meteorology, and healthcare, where they hold significant analytical and practical value. However, missing values caused by human errors or unforeseen events often compromise the completeness of time series, thereby undermining the accuracy and reliability of subsequent analysis and decision-making. Incomplete time series typically exhibit complex temporal dependencies and inter-variable correlations, making effective imputation a challenging task. Deep learning methods, with their powerful modeling capabilities, have emerged as a promising solution to address these challenges and have become an active area of research. We provide a comprehensive review of incomplete time series imputation methods based on deep learning. The survey begins by introducing the relevant concepts and definitions, followed by a systematic categorization and analysis of representative methods, highlighting their core ideas and characteristics. Commonly used open-source datasets and evaluation metrics are then summarized, along with a well-designed experimental framework. The experiments evaluate 10 mainstream deep learning models in terms of imputation performance and efficiency under various conditions, including different missing scenarios, missing rates, variable dimensions, sequence lengths, and downstream task performance. We analyze the experimental results to derive comprehensive insights, then recommend appropriate imputation methods based on these findings. Finally, we discuss current research trends and future directions in this field, concluding the survey.

Key words　deep learning; large model; time series; incomplete data imputation; data governance

摘　要　时序数据广泛存在于工业、金融、交通、气象及医疗等领域，具有重要的分析应用价值.然而由于人为或偶发因素导致的缺失，时序数据的完整性经常遭到破坏，进而削弱基于其开展的分析与决策的准确性和可靠性.不完整时序数据通常包含复杂的时序依赖和变量间关系，为缺失值的有效补全带来较大挑战.深度学习方法具有强大的建模能力，是应对该挑战的有效技术，并日益成为研究热点.故系统综述基于深度学习的不完整时序数据补全方法的研究现状，首先介绍其相关概念和定义，进而对现有方法进行分类，归纳各类代表性方法的核心思想与特点.随后整理常用的开源数据集与评价指标，并设计全面系统的实验方案.实验从缺失场景、缺失率、变量数量、序列长度以及对下游任务的提升效果等多个角度，评

收稿日期：2025-06-06；**修回日期**：2025-07-16

基金项目：浙江省“领雁”计划攻关课题（2024C0110902）

This work was supported by the Leading Goose Research and Development Program of Zhejiang (2024C0110902).

通信作者：苗晓晔（miaoxy@zju.edu.cn）

估 10 种主流深度学习方法的补全质量与效率,最终总结实验结果并依据场景推荐合适的补全方法.最后,结合当前研究进展对该领域的未来发展趋势进行展望,并总结全文.

关键词 深度学习;大模型;时间序列;不完整数据补全;数据治理

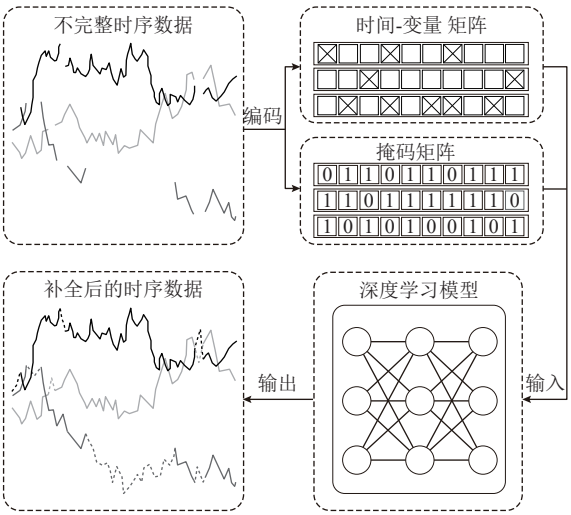
中图法分类号 TP183; TP274

DOI: 10.7544/issn1000-1239.202550449 **CSTR:** 32373.14.issn1000-1239.202550449

在如今的大数据时代,时间序列数据(time series data,下文简称“时序数据”)在工业[1]、金融[2]、交通[3]、气象[4]、医疗[5]以及健康[6]等各个领域普遍存在.然而在时序数据的收集、传输和存储等过程中,常伴随着诸多不确定因素,如停电、传感器故障、人为遗漏、隐私保护等原因均可能导致时序数据中出现缺失值.当这些不完整时序数据被应用至下游特定任务中时,数据中的缺失值可能会对时序数据分析带来严峻挑战,并影响决策的准确性与可靠性.例如 ICU 数据集 PhysioNet2012[7]的缺失率高达 80% 左右,使得在该数据集上进行的重要时序分析任务(预测病人入住 ICU 48 小时内是否会死亡)变得极具挑战.如何高效准确地对不完整时序数据进行补全(imputation),已成为时序数据分析领域的重要研究方向.

针对不完整时序数据中的缺失值问题,研究人员已提出多种类型的补全方法.早期的统计方法使用时序数据中的平均数、众数或最后观测值替换缺失值,或利用统计模型(如 ARIMA[8])对缺失值进行建模.随着计算机技术的发展,机器学习方法(如 K 最近邻[9]、矩阵分解[10]、回归树[11]等)被广泛应用于时序数据补全任务中.然而时序数据中同时存在复杂的时序依赖(temporal dependencies)和变量间关系,使得传统的统计方法或机器学习方法在处理这些结构复杂的数据时,其补全质量往往受到限制.

近年来,深度学习(deep learning)方法在不完整时序数据补全领域快速发展.这些方法基于基本的神经网络架构(如循环神经网络(RNN)[12]、卷积神经网络(CNN)[13]、Transformer[14]等)捕捉时间序列中的复杂依赖,并结合生成对抗网络(GAN)[15]、扩散模型[16]、图神经网络(GNN)[17]等深度学习技术,实现高质量的缺失值补全.在最新的研究工作中,基于海量数据预训练的大模型[18],在面对从未见过的不完整时序数据时亦表现出良好的补全性能,体现出其显著的泛化能力.图 1 展示了深度学习方法补全不完整时序数据的典型过程.本文依据深度学习方法在捕捉时序依赖过程中利用的基本神经网络架构,将现有的深度学习不完整时序数据补全方法划分为 4 类:基于 RNN、基于 CNN、基于注意力机制和基于大模型的方法.对于每一类方法,本文将介绍其中具有代表性的研究工作,并归纳其适用任务类型、输出类型及所采用的增强技术等关键特征.



注:图中的时序数据为具有 3 个变量的多变量时序数据.

Fig. 1 Process of incomplete time series imputation by deep learning methods

图 1 深度学习方法对不完整时序数据补全的过程

为对在实际任务中合理使用深度学习时序数据补全方法提供建议,本文设计了全面的实验来评估多种深度学习补全方法在不同缺失场景(包括缺失机制、缺失模式)、缺失率、变量数量和序列长度下的补全质量与效率,并分析补全结果对下游任务的影响.实验结果表明,与为通用时序任务设计的方法相比,专为补全任务开发的深度学习方法在多数数据集上具有更高的补全质量.补全方法的性能显著受到缺失机制和缺失模式的影响,例如在非随机缺失机制下以及时间与变量维度连续缺失的模式下,补全方法的性能明显下降.随着缺失率的增加以及变量数量或序列长度的变化,补全质量也呈现出一定程度的波动,且不同方法在不同数据集上对这些因素的敏感程度不同.尽管深度学习的补全结果通常优于传统方法,并在一定程度上提升了下游任务的性能,但补全误差的降低未必都能转化为下游任

务性能的提升. 此外, 实验总结了不完整时序数据中影响深度学习方法补全性能的典型特征, 并据此推荐合适的补全方法.

基于对现有方法的综述分析及实验结果的启示, 本文从以下方面对深度学习不完整时序数据补全方法的发展趋势进行展望: 面向大规模时序数据的补全方法、复杂缺失场景下的时序数据补全方法、时序数据补全与下游任务的集成优化、面向时序特性的损失函数与评价指标设计、大模型与多模态信息的融合应用、面向在线与轻量化部署场景的补全方法. 针对每个方向, 本文均提出了可行的研究思路与实现建议, 旨在为未来研究提供有价值的参考.

1 相关工作

针对时序数据补全任务, 目前已有多项综述工作从不同角度进行了总结. Khayati 等人[19] 综述了基于矩阵与模式的机器学习方法, 并进行了系统实验, 但并未涉及深度学习方法. 此外, 部分文献聚焦于特定类型的深度学习时序数据补全方法, 如 Fang 等人[20]、Weerakody 等人[21] 的工作分别综述了基于 RNN 和 GRU[22] 的方法, Kazijevs 等人[6] 则针对健康领域的方法展开讨论. 受研究视角与发表时间限制, 上述综述覆盖范围相对有限, 尚未形成统一的分类框架, 缺乏对深度学习时序数据补全方法发展的整体把握.

另一部分综述则围绕特定的深度学习模型在通用时序分析任务中的应用展开, 例如 Brophy 等人[23]、Jin 等人[24]、Jin 等人[25] 和 Lin 等人[26] 分别聚焦于 GAN、GNN、大模型及扩散模型的相关研究. 尽管这些综述在一定程度上包含了补全任务, 但补全仅为其中内容占比较小的一个子任务, 缺乏对补全方法的系统分类比较与实验验证.

Wang 等人[27] 从生成式与预测式的角度对深度学习时序数据补全方法进行分类, 并尝试构建统一框架, 涵盖了大部分深度学习方法, 具有一定的理论贡献. 然而, 该工作受限于发表时间, 尚未涵盖近年来兴起的大模型方法, 且对具体方法的技术细节分析不足, 缺乏实验层面的系统评估与实践指导.

表 1 对相关代表性文献进行了汇总与比较. 综上所述, 当前时序数据补全领域的综述工作仍存在以下不足: 1)缺乏覆盖主流深度学习方法的统一分类框架; 2)缺乏对不同方法间技术差异与关联的系统比较; 3)缺乏面向实际应用的实验评估与方法推荐. 随着大模型方法的迅速发展以及时序分析任务对补全质量的要求日益提升, 深度学习时序数据补全方法正呈现出快速演化趋势, 亟需一篇覆盖全面、分类清晰、深入评估并具有实际指导意义的综述. 本文正是在此背景下提出, 旨在系统梳理深度学习时序数据补全方法的研究进展, 统一分类比较各类方法, 并通过全面的实验评估为方法选择与应用提供有效参考.

Table 1 Comparison of Related Surveys on Time Series Imputation

表 1 时序数据补全相关综述对比

综述文章	年份	深度学习方法	大模型方法	实验设置	方法选择建议
Khayati 等人[19]	2020			●	●
Fang 等人[20]	2020	○			
Weerakody 等人[21]	2021	○		○	
Kazijevs 等人[6]	2023	○		●	●
Brophy 等人[23]	2023	○			
Jin 等人[25]	2023		●		
Jin 等人[24]	2024	○			
Lin 等人[26]	2024	○			
Wang 等人[27]	2024	●		○	
本文	2025	●	●	●	●

注: ●代表全面的综述、实验或建议; ○代表部分涉及.

2 相关概念与定义

本节将介绍时序数据的概念及特点, 进一步给出不完整时序数据补全的核心定义并结合实例进行说明, 最后概述缺失机制的相关概念.

2.1 时序数据

时序数据是指按照时间顺序有序排列的观测(observation)数据. 用 $D=\{(\boldsymbol{X}_1,\boldsymbol{T}_1),(\boldsymbol{X}_2,\boldsymbol{T}_2),\cdots,(\boldsymbol{X}_N,\boldsymbol{T}_N)\}$ 表示一个包含 N 个时序数据样本的时序数据集, 样本 $\boldsymbol{X}=(\boldsymbol{\mathcal{X}}_1,\cdots,\boldsymbol{\mathcal{X}}_i,\cdots,\boldsymbol{\mathcal{X}}_n)^{\mathrm{T}}\in\mathbb{R}^{n\times d}$ 与时间序列 $\boldsymbol{T}=(t_1,\cdots,t_i,\cdots,t_n)^{\mathrm{T}}\in\mathbb{R}^n$ 相对应, $\boldsymbol{\mathcal{X}}_i$ 表示 $\boldsymbol{X}$ 在时间步 t_i 上的观测, 每个观测包含 d 个变量的观测值, 可被表示为 $\boldsymbol{\mathcal{X}}_i=(x_i^1,\cdots,x_i^j,\cdots,x_i^d)$. 例如 x_i^j 表示时序数据 $\boldsymbol{X}$ 中的第 j 个变量在时间步 t_i 上的观测值. 当 $d=1$ 时, 称 $\boldsymbol{X}$ 为单变量时序数据; 当 $d\geqslant 2$ 时, 称 $\boldsymbol{X}$ 为多变量时序数据.

与一般的表格数据不同, 时序数据中的各个观测值有其顺序性, 打乱顺序会破坏其内在结构. 此外, 时序数据通常存在时序依赖性, 即当前时间步的观测值依赖于过去(或未来)的观测值, 这种依赖可以是短期的或者长期的. 另外, 现实世界中的时序数据往往还包含复杂的趋势项(trend)、季节项(seasonality)、

周期项(cycle)与噪声项(noise)[28], 这些成分在提供分析信息的同时也增加了建模难度.

2.2 不完整时序数据

若$\boldsymbol{X}$中存在部分未观测到的缺失值, 即某些位置的观测值不可用, 则称其为不完整时序数据, 记$\boldsymbol{X}^{\text{obs}}$为$\boldsymbol{X}$中观测到的部分, $\boldsymbol{X}^{\text{mis}}$为缺失部分.

定义 1. 掩码矩阵.

为编码$\boldsymbol{X}$中的缺失情况, 引入一个掩码矩阵(mask matrix), $\boldsymbol{M}=(\boldsymbol{m}_1,\cdots,\boldsymbol{m}_i,\cdots,\boldsymbol{m}_n)\in\mathbb{R}^{n\times d}$, 其中$\boldsymbol{m}_i$表示$\boldsymbol{X}$在时间步$t_i$上的缺失情况. 进一步地, $\boldsymbol{m}_i=(m_i^1,\cdots,m_i^j,\cdots,m_i^d)$, m_i^j与x_i^j一一对应, 表示$\boldsymbol{X}$中的某个变量在某个时间步上是否缺失, 当x_i^j可以被观测到时, $m_i^j=1$, 当x_i^j缺失时, $m_i^j=0$.

定义 2. 时滞矩阵.

时滞矩阵(lag matrix)$\boldsymbol{\delta}\in\mathbb{R}^{n\times d}$被用于表示某一变量在当前时间步上距离该变量的上一个非缺失位置的时间跨度, 其计算方式为:

$$\delta_i^j=\begin{cases}t_i-t_{i-1}, & m_{i-1}^j=1,\\ \delta_{i-1}^j+t_i-t_{i-1}, & m_{i-1}^j=0\ \text{且}\ i>0,\\ 0, & i=0.\end{cases}\tag{1}$$

示例 1.

现给定一个时序数据样本$\boldsymbol{X}$(缺失值用NaN表示)及其对应的时间序列$\boldsymbol{T}$:

$$\boldsymbol{X}=\begin{pmatrix}\text{NaN}&8&6&10\\6&7&\text{NaN}&\text{NaN}\\\text{NaN}&10&\text{NaN}&9\end{pmatrix},\ \boldsymbol{T}=\begin{pmatrix}0\\3\\8\end{pmatrix},\tag{2}$$

则其掩码矩阵$\boldsymbol{M}$和时滞矩阵$\boldsymbol{\delta}$分别为:

$$\boldsymbol{M}=\begin{pmatrix}0&1&1&1\\1&1&0&0\\0&1&0&1\end{pmatrix},\ \boldsymbol{\delta}=\begin{pmatrix}0&0&0&0\\3&3&3&3\\5&5&8&8\end{pmatrix}.\tag{3}$$

2.3 不完整时序数据补全

基于定义 1 和定义 2, 将一个不完整时序数据样本$\boldsymbol{X}$编码为$\{\boldsymbol{X}^{\text{obs}},\boldsymbol{M},\boldsymbol{\delta}\}$. 不完整时序数据补全的任务是学习一个参数为$\theta$的模型$\mathcal{M}_\theta$, 利用$\boldsymbol{X}$的可观测部分重构完整时序数据$\hat{\boldsymbol{X}}$, 该过程可用下式表示:

$$\hat{\boldsymbol{X}}=\boldsymbol{M}\odot\boldsymbol{X}^{\text{obs}}+(1-\boldsymbol{M})\odot\bar{\boldsymbol{X}},\tag{4}$$

其中$\odot$表示矩阵的哈达玛积(按元素相乘), $\bar{\boldsymbol{X}}=\mathcal{M}_\theta(\boldsymbol{X}^{\text{obs}},\boldsymbol{M},\boldsymbol{\delta})$, 即$\bar{\boldsymbol{X}}$由$\mathcal{M}_\theta$根据$\boldsymbol{X}$中的信息重建得到. 不完整时序数据补全的目标主要包括 2 点: 1)补全后的时序数据$\hat{\boldsymbol{X}}$应尽可能接近理想的完整时序数据$\boldsymbol{X}^{\text{com}}$; 2)补全后的时序数据$\hat{\boldsymbol{X}}$在下游任务中的使用效果应优于原始时序数据$\boldsymbol{X}$[27].

2.4 缺失机制

缺失机制描述了数据是否缺失与其本身之间的关联关系, 即$\boldsymbol{M}$与$\boldsymbol{X}$之间的关联, Van 等人[29]将这种关联称为缺失数据模型(missing data model), 其一般形式为$Pr(\boldsymbol{M}|\boldsymbol{X}^{\text{obs}},\boldsymbol{X}^{\text{mis}},\psi)$, 其中$\psi$表示缺失模型的参数. Rubin[30]将不完整数据中的缺失机制分为 3 类:

1)完全随机缺失(missing completely at random, MCAR). 所有数据点的缺失概率相同, 且该概率与数据本身无关. 尽管 MCAR 假设便于建模, 但现实数据中较少满足. MCAR 缺失模型可以被表示为:

$$Pr(\boldsymbol{M}|\boldsymbol{X}^{\text{obs}},\boldsymbol{X}^{\text{mis}},\psi)=Pr(\boldsymbol{M}|\psi).\tag{5}$$

2)随机缺失(missing at random, MAR). 缺失概率可能依赖于部分观测数据(而非缺失值本身). 与 MCAR 假设相比, MAR 假设更贴近现实场景, 也是大多数现代缺失值处理方法所依据的前提[29]. MAR 缺失模型可以被表示为:

$$Pr(\boldsymbol{M}|\boldsymbol{X}^{\text{obs}},\boldsymbol{X}^{\text{mis}},\psi)=Pr(\boldsymbol{M}|\boldsymbol{X}^{\text{obs}},\psi).\tag{6}$$

3)非随机缺失(missing not at random, MNAR). 缺失概率与缺失值本身相关, 即无法通过可观测数据完全解释. MNAR 是最复杂的情况, MCAR 和 MAR 可视为其特例. MNAR 缺失模型的形式无法被化简:

$$Pr(\boldsymbol{M}|\boldsymbol{X}^{\text{obs}},\boldsymbol{X}^{\text{mis}},\psi).\tag{7}$$

当数据的缺失机制满足 MCAR 或 MAR 假设时, 可认为缺失模型是“可忽略的”[29], 即$\mathcal{M}_\theta$只需专注学习$\boldsymbol{X}^{\text{obs}}$的分布. 而在 MNAR 假设下, 缺失机制是“不可忽略的”, $\mathcal{M}_\theta$需要同时建模$\boldsymbol{X}^{\text{obs}}$的分布和缺失模型, 否则将导致补全结果的偏差. 值得注意的是, 当前多数基于深度学习的不完整时序数据补全方法假设数据满足 MCAR, 部分方法考虑了 MAR, 极少数研究尝试处理 MNAR 假设下的时序数据.

3 不完整时序数据补全方法分类

现有处理时序数据缺失问题的方法大致可分为删除法、统计方法、机器学习方法和深度学习方法 4 类. 表 2 列举了各分类中的代表性方法, 下文将简要介绍各类方法的核心思想与典型工作.

3.1 删除法

删除法[29]是处理不完整时序数据最基础的策略, 主要包括成列删除(listwise deletion)和成对删除(pairwise deletion). 成列删除直接剔除包含缺失值的样本, 适用于缺失率较低的情况, 但在缺失率较高时会显著减少可用数据样本, 影响分析结果. 成对删除则仅在特定分析中排除不完整样本, 可最大程度利用已有数据, 但不同分析可能基于不同样本子集, 易

Table 2 Categories of Incomplete Time Series Imputation Methods

表 2 不完整时序数据补全方法分类

分类	技术路线	方法
删除法	删除法	Listwise Deletion[29], Pairwise Deletion[29]
统计方法	基于统计量	Mean, Median, LOCF[31]
	基于回归	ARIMA[8], IMR[32]
	基于多重插补	MICE[33], IIM[34]
机器学习方法	基于相似性	TKCM[35], ST-MVL[36]
	基于树	DMI[37], SMILES[38]
	基于矩阵	SoftImpute[39], TRMF[40],
深度学习方法	基于 RNN	GRU-D[41], BRITS[42], GRUI-GAN[43], E^{2}GAN[44], NAOMI[45], SSGAN[46], GRIN[17]
	基于 CNN	GP-VAE[47], TimesNet[48], ModernTCN[49]
	基于注意力机制	DeepMVI[50], CSDI[51], SPIN[52], PriSTI[53], SAITS[54], Peri-midFormer[55], ImputeFormer[56], FGTI[57], Diffusion-TS[58], UP2ME[59], TimeMixer++[60]
	基于大模型	GPT4TS[61], LLM-TS Integrator[62], MOMENT[63], Timer[64], TOTEM[65]

引发结果不一致，尤其在小样本情境下更为明显.

3.2 统计方法

基于统计量的方法通过均值(mean)、中位数(median)等简单统计量补全缺失值，计算高效但忽略了时序结构，难以应对波动较大的数据；LOCF(last observation carried forward)[31]是一种更符合时序数据特点的统计方法，该方法用缺失位置之前最近的观测值替换缺失值，可保持局部连续性，适用于短期缺失，但对长期缺失或剧烈波动的数据效果较差. 基于回归的方法通过构建回归模型对缺失值进行估计：经典时序分析模型 ARIMA[8]利用自回归、差分与移动平均建模时间序列，适合平稳数据；IMR [32]则通过迭代最小修复策略同时处理异常与缺失. 多重插补方法通过多轮建模提升补全准确度：MICE[33]为每个缺失变量单独建模并迭代优化；IIM[34]则为每个缺失样本构建专门的模型，能处理异质和稀疏的数据.

基于统计的时序数据补全方法实现简单且具有良好的解释性，但难以捕捉时序数据中的时序依赖关系，在复杂时序数据中的补全质量不佳[27].

3.3 机器学习方法

基于相似性的方法通过 K 最近邻(k-nearest neighbors, KNN)算法[9]或其他聚类方法寻找相似样本，参考这些相似样本补全缺失部分：TKCM[35]从参考序列中匹配最相似的模式，利用其观测估计缺失数据；ST-MVL[36]结合空间和时间 2 个维度的相似性，通过协同过滤和局部平滑来预测缺失值. 基于树的方法利用决策树[11]或其他更先进的树模型建模时序依赖：DMI[37]通过完整数据训练决策树后用于缺失值补全；SMILES[38]在初步补全后，提取时序特征并借助 XGBoost[66]模型进行监督学习预测缺失值. 基于矩阵的方法将时序数据表示为时间-变量矩阵，通过降维和重构完成补全：SoftImpute[39]基于核范数最小化思想，通过迭代式奇异分解的方法完成低秩矩阵补全；TRMF[40]则通过在矩阵分解框架中引入时间正则项，有效建模数据中的时序依赖.

机器学习方法通过学习时序数据的内在结构进行缺失值补全，部分方法能够在较强表达能力与计算效率间取得平衡，适用于大规模时序数据. 然而这些方法的补全效果依赖于特定的数据结构假设(如低秩性、平稳性、线性可分等)，在实际应用中可能面临泛化能力不足的问题. 例如基于矩阵分解的方法在处理部分时间步上所有变量均缺失时会面临挑战[19]. 在存在大量缺失或噪声、非线性以及长期依赖等场景下，机器学习方法的表现受限.

3.4 深度学习方法

随着深度学习的发展，越来越多的方法将神经网络用于不完整时序数据补全. 这类方法通过大规模参数建模数据中的时序依赖，具有更强的表达能力与泛化能力. 多数深度学习方法试图学习一个建模历史(或未来)信息的函数从而对缺失值进行估计. 在最简单的状况下(在单变量时序数据上，使用缺失位置之前的信息补全缺失值，此处将x_i^j简写为x_i)，时间步t_i上的补全值$\hat{x}_i$可被表达为：

$$\hat{x}_i = \mathcal{M}(x_{i-\tau:i}, m_{i-\tau:i}, \delta_{i-\tau:i}), \tag{8}$$

其中$\mathcal{M}(\cdot)$为通过已知信息学习到的模型，可利用相关观测预测缺失值，τ为参与预测的数据的窗口长度，$x_{i-\tau:i}=\{x_{i-\tau},\cdots,x_{i-1}\}$表示当前缺失值之前的$\tau$个时间步的数据，同理，$m_{i-\tau:i}=\{m_{i-\tau},\cdots,m_{i-1}\}$，$\delta_{i-\tau:i}=\{\delta_{i-\tau},\cdots,\delta_{i-1}\}$.

深度神经网络通过一系列非线性层构建中间表征表示来学习预测关系[67]，在不完整时序数据补全领域中，这可以被看作是将相关的时序数据信息编码到一个隐状态z_i中，最终的补全结果仅由z_i决定：

$$z_i = g_{\text{enc}}(x_{i-\tau:i}, m_{i-\tau:i}, \delta_{i-\tau:i}), \tag{9}$$

$$\hat{x}_i = g_{\text{dec}}(z_i), \tag{10}$$

其中g_{enc}和g_{dec}分别表示编码器(encoder)和解码器(decoder)[68]，用于将输入的时序信息编码后通过一系列非线性层再解码得到补全结果. 当前的深度学

习方法主要利用3种类型的基础神经网络架构作为编码器提取时序数据中的信息: RNN[12]、CNN[13]、注意力机制(attention mechanism)[14]. 神经网络架构的选择决定了模型能从原始数据中捕捉何种时序依赖关系, 并据此完成缺失值补全.

3.4.1 RNN

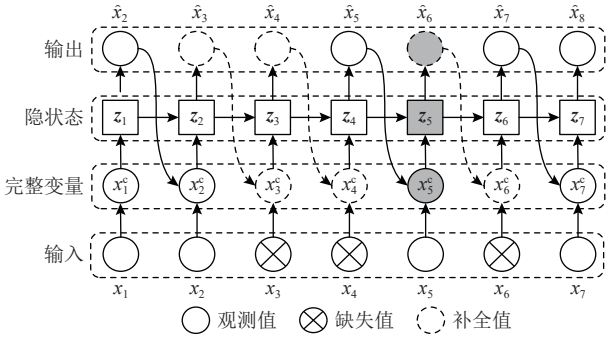
由于时序数据的序列属性, RNN是最早应用于时序数据补全的深度神经网络之一. RNN的核心思想是将当前时间步数据x_i以及过去的信息压缩为一个隐状态z_i, 每个时间步的隐状态z_i由当前时间步的输入信息对上一步的隐状态z_{i-1}进行更新得到. 图2示意了一个简单的单变量单向RNN时序数据补全模型的工作过程, 这个模型使用上一时间步所更新的隐状态z_{i-1}来得到当前时间步的补全值$\hat{x}_i$:

$$\hat{x}_i = \boldsymbol{W}_x \boldsymbol{z}_{i-1} + b_x, \tag{11}$$

其中$\boldsymbol{W}_x$和b_x为模型参数, 而当前时间步的隐状态$\boldsymbol{z}_i$可以被表达为:

$$\boldsymbol{z}_i = \sigma(\boldsymbol{W}_z \boldsymbol{z}_{i-1} + \boldsymbol{U}_z x_i^c + \boldsymbol{b}_z), \tag{12}$$

其中σ表示激活函数(可以是sigmoid函数或其他非线性函数), $\boldsymbol{W}_z$, $\boldsymbol{U}_z$和$\boldsymbol{b}_z$为模型参数, $x_i^c = m_i x_i + (1-m_i)\hat{x}_i$, 当$x_i$可被观测到时$x_i^c = x_i$, 当$x_i$缺失时用上一步的预测值$\hat{x}_i$代替$x_i$作为输入.



注: 图中阴影部分示意了将缺失值x_6补全为$\hat{x}_6$的过程.

Fig. 2 Imputation process of a unidirectional RNN model

图2 一个单向RNN模型的补全过程

3.4.2 CNN

CNN最早被设计用于图像处理领域, BAI等人[69]将其迁移到时序数据领域并提出时序卷积网络(temporal convolutional network, TCN). TCN遵循2个原则: 1) TCN的输入和输出序列长度相同; 2) TCN只能利用当前时间步之前的信息. TCN采用一维全卷积网络(fully-convolutional network, FCN)[70]结合因果卷积(仅利用当前时间步及其之前的元素进行卷积操作)的方式在时序数据处理中保证上述2点原则. 在给定输入的数据为单变量时间序列$\boldsymbol{X} = (x_1, \cdots, x_n) \in \mathbb{R}^n$和卷积核$f: \{0, \cdots, k-1\} \to \mathbb{R}$的情况下, 单层卷积操作可以由式(13)表示, 其中$*$表示卷积操作, $\boldsymbol{b}$表示偏置, σ表示激活函数, $\boldsymbol{Z} = \{\boldsymbol{z}_1, \cdots, \boldsymbol{z}_n\} \in \mathbb{R}^{n \times d_z}$为经过卷积操作$\mathcal{F}(\cdot)$后的隐变量序列, d_z为隐变量维度. 当存在多个卷积层时, 输入$\boldsymbol{X}$经过l个卷积层得到的隐变量序列$\boldsymbol{Z}^{(l)}$可由式(14)表示.

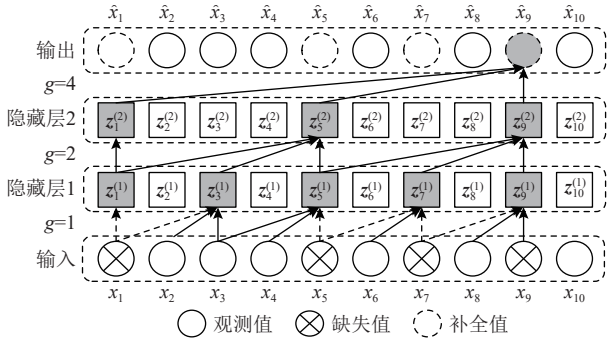
$$\boldsymbol{Z} = \mathcal{F}(\boldsymbol{X}) = \sigma(f * \boldsymbol{X} + \boldsymbol{b}), \tag{13}$$

$$\boldsymbol{Z}^{(l)} = \mathcal{F}^{(l)} \circ \mathcal{F}^{(l-1)} \circ \cdots \circ \mathcal{F}^{(1)}(\boldsymbol{X}). \tag{14}$$

为充分捕捉长期依赖, TCN使用了膨胀卷积(dilated convolution)扩大在预测时的感受野, 在当前时间步t_i上的膨胀卷积操作可以被定义为:

$$\mathcal{F}(i) = (x *_g f)(i) = \sum_{c=0}^{k-1} f(c) \cdot x_{i-g \cdot c}, \tag{15}$$

其中g表示膨胀因子, 表示参与卷积操作的2个邻近元素之间的间隔, k表示卷积核的大小, 即参与卷积操作的元素数量. 图3示意了一个TCN时序数据补全模型的工作过程, 这个模型的膨胀卷积因子为$g \in \{1, 2, 4\}$, 卷积核大小为3, 原始输入数据$\boldsymbol{X}$在经过3层卷积操作后得到其长度相同的预测数据$\hat{\boldsymbol{X}}$.



注: 图中阴影部分示意了将缺失值x_9补全为$\hat{x}_9$的过程.

Fig. 3 TCN imputation process with causal and dilated convolutions

图3 采用因果卷积和膨胀卷积的TCN的补全过程

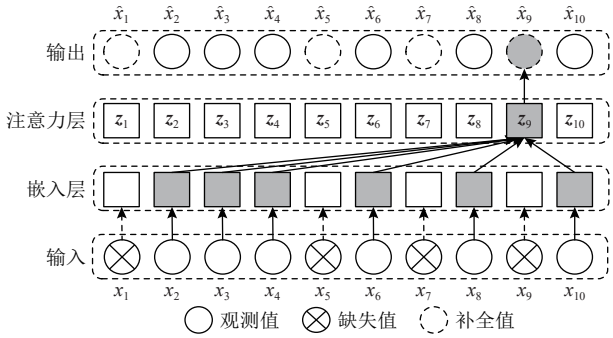
3.4.3 注意力机制

Transformer[14]的提出使得注意力机制在自然语言处理领域取得突破性进展, 最新提出的深度学习不完整时序数据补全方法大都使用注意力机制建模时序依赖. 注意力机制允许模型在同一时间感知整个时间序列的信息, 通过计算注意力分数来赋予关键信息更大的权重, 使其能够有效处理长距离的时序依赖. 注意力机制将输入的数据$\boldsymbol{X}$分别嵌入为查询($\boldsymbol{Q}$)、键($\boldsymbol{K}$)和值($\boldsymbol{V}$), 通过计算$\boldsymbol{Q}$与$\boldsymbol{K}$之间的距离, 赋予与$\boldsymbol{Q}$更接近的$\boldsymbol{K}$更大的权重, 权重与对应的$\boldsymbol{V}$相

乘后得到注意力层的输出，该过程可由下式表示：

$$\text{Attention}(\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V}) = softmax\left(\frac{\boldsymbol{Q}\boldsymbol{K}^{\mathrm{T}}}{\sqrt{d_{\boldsymbol{K}}}}\right)\boldsymbol{V}, \tag{16}$$

其中$d_{\boldsymbol{K}}$为$\boldsymbol{K}$的维度，*softmax*函数确保所有权重之和为1. 注意力层的输出随后会被输入到全连接层以生成补全结果. 原始的Transformer使用自注意力机制，即查询、键和值均由同一时间序列输入嵌入得到. 图4示意了一个基于注意力机制的模型在单步中聚合所有时间步信息的过程，随后模型使用聚合后的信息完成对缺失值的补全.



注：图中阴影部分示意了将缺失值x_9补全为$\hat{x}_9$的过程.

Fig. 4 Imputation process of attention mechanism-based model

图4 注意力机制模型的补全过程

3.4.4 大模型

近年来，大模型成为时序数据补全研究的新兴方向，相关方法应用的大模型主要可分为预训练基座模型(pre-trained foundation model, PFM)[25]和大语言模型(large language model, LLM)[18]. 这些方法大多基于Transformer架构，并通过大规模数据预训练获得通用建模能力，能在不同种类的任务中展现出良好的泛化性能[71].

时序领域的PFM专为时序数据设计，具备处理多种时序分析任务(补全、预测、分类及异常检测等)的通用能力. 在构建PFM时，需依赖涵盖多领域、多采样频率、多变量维度的大规模异构时序数据集作为训练语料. 为统一处理不同结构的输入，PFM通常对时序数据进行分段(patch)或者词元化(tokenize)，从而适配统一的模型架构. 在面对多变量时序数据时，目前多数PFM会将其拆分为若干单变量序列进行预训练，并在后续具体任务中使用微调等方式建模变量间关系. PFM通常具备少样本(few-shot)和零样本(zero-shot)预测能力，无需从头开始训练就能快速适配具体时序任务.

另一类方法则尝试将LLM直接迁移到时序领域中. LLM可被视为PFM中的一类，其最初在大规模文本数据集上训练，具备强大的上下文建模和泛化能力. 目前用于时序分析的LLM可被分为可见嵌入大语言模型(embedding-visible LLMs)和不可见嵌入大语言模型(embedding-invisible LLMs)[25].

可见嵌入大语言模型通常是开源的(如LLaMA[72]等). 研究人员通过微调(如LoRA、adapter等)[25]使LLM适配时序任务. 这些方法往往仅调整LLM的部分参数以保持模型的生成能力并节省计算开销.

不可见嵌入大语言模型通常是闭源的(如GPT-4[73]等)，主要通过API接口提供服务. 基于这类LLM的方法在使用中将时序数据转换为提示词(prompt)，再输入至LLM中进行推理预测. 这种方式无需修改模型本身就可以充分利用LLM的生成能力.

4 基于深度学习的不完整时序数据补全方法

本节依据捕捉时序依赖的基础架构，将现有深度学习时序数据补全方法分为4类：基于RNN、基于CNN、基于注意力机制、基于大模型的方法. 需要指出的是尽管基于大模型的方法一般而言也是基于注意力机制捕捉时序依赖，但由于其使用大量数据进行预训练并在具体任务上微调的策略与传统基于注意力的方法显著不同，故将其单独作为一类讨论. 表3对各分类中具有代表性的方法进行了归纳整理，后文将对这些方法的技术路线、适用的时序任务类型、输出类型、使用的增强技术等进行介绍.

4.1 基于RNN的方法

Che等人[41]提出的GRU-D(gated recurrent unit with decay)方法是较早将RNN应用于时序数据补全的代表作. 该方法在门控循环单元(gated recurrent unit, GRU)[22]的基础上引入了衰减机制，用以建模缺失值与观测值之间的时间距离影响. 其核心假设包括：1)当变量长时间未被观测时，其值趋近于某一默认水平；2)已有观测值对缺失值的影响会随时间递减. 通过引入时间衰减因子，GRU-D能更有效地估计缺失值，缓解传统RNN在长序列建模中信息遗失的问题.

Cao等人[42]提出的BRITS(bidirectional recurrent imputation for time series)方法可以同时利用当前时间步之前和之后的信息进行补全. 该方法构建了正向与反向2个RNN分支，通过一致性损失项确保2个方向预测的一致性. 同时，当变量间存在相关性时，BRITS引入了变量间交叉估计机制，融合基于时序维度与变量维度的补全结果，从而增强模型对复杂依赖关系的建模能力.

Table 3 Incomplete Time Series Imputation Methods Based on Deep Learning

表 3 基于深度学习的不完整时序数据补全方法

分类	方法	发表	时序任务类型	输出类型	增强技术
基于 RNN 的方法	GRU-D[41]	Sci. Rep.'2018	补全、分类	数值	GRU
	BRITS[42]	NeurIPS'2018	补全、分类	数值	双向 RNN
	GRUI-GAN[43]	NeurIPS'2018	补全	数值	GAN
	E2GAN[44]	IJCAI'2019	补全	数值	GAN，AE
	NAOMI[45]	NeurIPS'2019	补全	数值	双向 RNN，GAN
	SSGAN[46]	AAAI'2021	补全	数值	双向 RNN，GAN
	GRIN[75]	ICLR'2022	补全	数值	GNN
基于 CNN 的方法	GP-VAE[47]	AISTATS'2020	补全	概率分布	VAE，高斯过程
	TimesNet[48]	ICLR'2023	通用任务	数值	FFT，Inception
	ModernTCN[49]	ICLR'2024	通用任务	数值	DWConv，ConvFFN
基于注意力机制的方法	DeepMVI[50]	VLDB'2021	补全	概率分布	CNN，核回归
	CSDI[51]	NeurIPS'2021	补全	概率分布	CNN，扩散模型
	SPIN[52]	NeurIPS'2022	补全	数值	GNN
	PriSTI[53]	ICDE'2023	补全	概率分布	GNN，扩散模型
	SAITS[54]	ESWA'2023	补全	数值	DMSA
	Peri-midFormer[55]	NeurIPS'2024	通用任务	数值	FFT
	ImputeFormer[56]	KDD'2024	补全	数值	FFT
	FGTI[57]	NeurIPS'2024	补全	概率分布	FFT，扩散模型
	Diffusion-TS[58]	ICLR'2024	补全、预测	概率分布	FFT，扩散模型
	UP2ME[59]	ICML'2024	通用任务	数值	预训练
	TimeMixer++[60]	arXiv'2024	通用任务	数值	FFT，CNN
基于大模型的方法	GPT4TS[61]	NeurIPS'2023	通用任务	数值	大语言模型微调
	LLM-TS Integrator[62]	NeurIPS'2024	通用任务	数值	通过大语言模型对深度学习时序模型进行微调
	MOMENT[63]	PMLR'2024	通用任务	数值	预训练
	Timer[64]	ICML'2024	通用任务	数值	预训练
	TOTEM[65]	TMLR'2024	通用任务	数值	预训练

Luo 等人[43]在 GRU-D 基础上引入了生成对抗网络(generative adversarial network, GAN)[15]，提出 GRUI-GAN. 该方法使用 GRU 构建生成器(generator)与判别器(discriminator)，以对抗方式优化缺失值的补全质量，并采用 WGAN 框架提升训练稳定性. 然而由于生成器使用随机噪声作为输入，其生成的数据质量存在限制. 因此，Luo 等人[44]进一步提出 E^{2}GAN 方法，引入自动编码器(auto-encoder, AE)[74]模块，学习输入序列的低维表示作为生成器的输入，从而提升生成器输出的针对性与精度，显著改善补全效果.

Liu 等人[45]提出的 NAOMI(non-autoregressive multiresolution imputation)方法与 BRITS 类似，同样采用双向 RNN 架构. NAOMI 通过正向和反向编码器提取时间序列的隐状态，其解码器则采用分治策略进行递归地补全，它首先进行粗粒度的补全，随后根据得到的补全结果进行更细粒度的补全，进而层层递进完成所有缺失值的补全任务. NAOMI 同样借助 GAN 框架以提升生成结果的分布一致性与质量.

Miao 等人[46]提出的 SSGAN(semi-supervised GAN)方法将半监督机制引入时序补全任务中. 如图 5 所示，在传统 GAN 框架的基础上，SSGAN 增加了分类器组件，通过联合优化时序补全与分类任务提升补全质量. 分类器利用少量有标签数据，对生成器输出的补全序列进行分类预测，并将分类误差反向传播至生成器，确保补全值不仅符合真实数据分布，还能提升下游分类任务的性能. 同时，该方法还设计了时间提醒矩阵，引导判别器专注于补全区域的数据质量.

Cini 等人[75]提出的 GRIN(graph recurrent imputation network)将图神经网络(graph neural network, GNN)[17]应用到了时序数据补全过程中以建模时序数据中可能存在的空间依赖(如采集自地理位置相近的气象传感器的数据在数值上可能更接近). 如

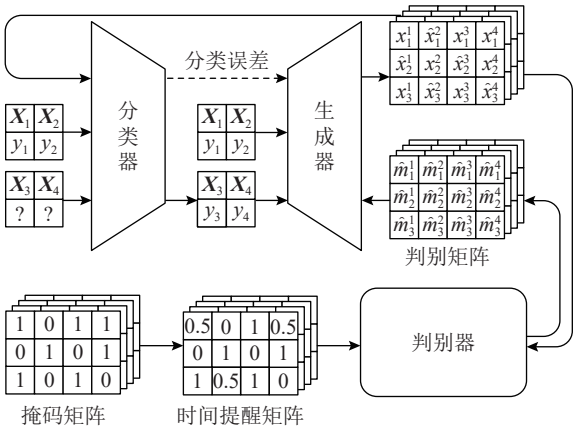


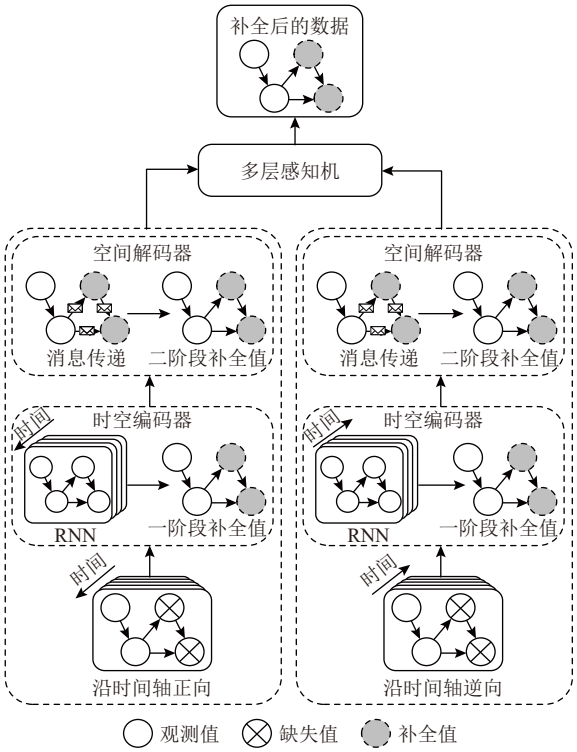
Fig. 5 Model architecture of SSGAN[46]

图 5 SSGAN 的模型结构[46]

图 6 所示，GRIN 将空间依赖建模为图结构，并采用双向两阶段的补全策略. 该方法首先通过由双向 RNN 构成的时空编码器获得补全值的初步估计，再利用空间解码器的消息传递层进行精细化调整，最后利用多层感知机(multilayer perceptron, MLP)[76] 融合双向信息生成最终补全结果.

4.2 基于 CNN 的方法

Fortuin 等人[47] 提出的 GP-VAE 方法融合了变分自编码器(variational autoencoders, VAE)[77] 与高斯过程(Gaussian process, GP)[78]，在隐变量空间中对时序数据进行建模. 其编码器采用 CNN 网络从带有缺失值的输入序列中提取观测值对应的隐变量分布，缺失值则通过固定值填充处理. 在隐空间中，隐变量序列被建模为服从柯西核(Cauchy kernel)的高斯过程，从而实现对时序结构的建模. 在重构阶段，该方法使用多层感知机将隐变量映射回原始空间，输出补全值的概率分布.



注：图中的动态图序列含有 4 个图节点，其中 2 个存在缺失.

Fig. 6 Model architecture of GRIN[75]

图 6 GRIN 的模型结构[75]

Wu 等人[48] 提出的 TimesNet 是一种通用时序建模架构，适用于补全、预测、分类、异常检测等任务. 其核心设计理念在于揭示时间序列中的多周期性结构. TimesNet 首先使用快速傅里叶变换(fast Fourier transform, FFT)提取时间序列中的主频，并据此构造多个候选周期，将一维时间序列折叠为多个二维时序图像. 随后，借助计算机视觉中成熟的 CNN 模块 Inception[79]，该方法在二维结构上联合建模周期内与周期间的模式变化. 原始时序输入在通过多个堆叠的能够加权融合不同时间尺度特征的 TimesBlock 模块后，输出最终的补全结果，如图 7 所示.

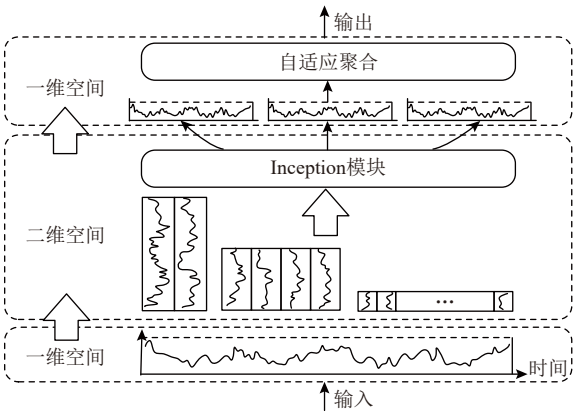


Fig. 7 TimesBlock in TimesNet[48]

图 7 TimesNet 中的 TimesBlock[48]

Luo 等人[49] 提出的 ModernTCN 方法并没有选择 TimesNet 中将时序数据按显式周期折叠的方式，而是在传统 TCN 架构基础上通过结构优化实现对隐式周期与长期依赖的建模. ModernTCN 由多个 ModernTCN Block 堆叠而成，每个 Block 基于深度卷积(DWConv)和卷积前馈网络(ConvFFN)两类模块构建. DWConv 使用大卷积核独立建模单变量时序依赖，不进行跨变量交互，ConvFFN 则负责变量间表征的交互建模，其中 ConvFFN1 提取每个变量的新表征，ConvFFN2 捕捉这些表征间的依赖关系. 这种模块化设计显著提升了模型的建模清晰度与泛化能力.

4.3 基于注意力机制的方法

Bansal 等人[50] 提出的 DeepMVI 将 Transformer 架

构与核回归相结合，构建了一个多模块、多尺度的补全框架. 模型由局部滑动窗口模块、时序 Transformer 模块以及核回归模块 3 部分组成，分别捕捉短期依赖、全局趋势和变量间相关性. 最终 3 路输出通过拼接并进行线性变换，输出缺失值的预测分布. 该方法的训练过程中引入自监督掩码与对抗训练策略，以提升泛化能力和输出分布的一致性.

Tashiro 等人[51]提出的 CSDI 方法通过引入条件扩散模型(conditional diffusion model)[16]用于建模缺失值的概率分布，不仅补全缺失值，还量化了其不确定性. 模型使用时间与变量双维度的自注意力模块进行建模. 在训练阶段，该方法将观测值随机划分为条件观测值(可见)和模拟缺失值，通过最小化补全结果与模拟缺失值的差距，模型可以学习到从噪声到真实值的条件分布. 在补全阶段，模型通过反向扩散迭代逐步从噪声中恢复缺失值，生成补全的概率分布.

Marisca 等人[52]提出的 SPIN 方法可视作 GRIN 的注意力机制替代版本. 该方法以图注意力结构替代双向 RNN，解决 GRIN 在稀疏观测下误差累积与计算效率问题. SPIN 将时间序列视作动态图信号，使用稀疏时空注意力模块聚合节点自身的时序信息与邻居节点的空间信息，最终输出补全值.

Liu 等人[53]提出的 PriSTI 方法综合了 CSDI 的条件扩散建模与 SPIN 的图注意力机制. 与 CSDI 从噪声中恢复补全值不同，PriSTI 首先通过线性插值生成的初步补全值作为更加稳定的初始条件. 随后结合时空图结构构造条件特征，通过堆叠的注意力层提取全局和局部时空信息作为条件，引导模型进行分阶段的去噪过程，最终输出补全的概率分布.

Du 等人[54]提出的 SAITS 针对原始 Transformer 架构对当前时间步的过度依赖问题，提出了对角线掩码自注意力(DMSA)机制，将注意力权重矩阵的对角线设为负无穷以强制模型关注其他时间步以及变量间关联等信息. 如图 8 所示，SAITS 模型中的 2 组 DMSA 模块分别对应 Transformer 中的编码器与解码器结构，通过权重融合模块将 2 组 DMSA 学习到的特征进行加权融合. 该方法采用随机掩码自监督训练策略，提升了其在真实数据集上的补全效果.

Wu 等人提出的[55]Peri-midFormer 方法引入特殊设计的金字塔注意力机制对时序数据进行建模. 该方法首先通过类似于 TimesNet 方法中的快速傅里叶变换分解周期成分，按照周期从长到短将时序数据分解为多组节点，并从上至下构造类似于树的金字

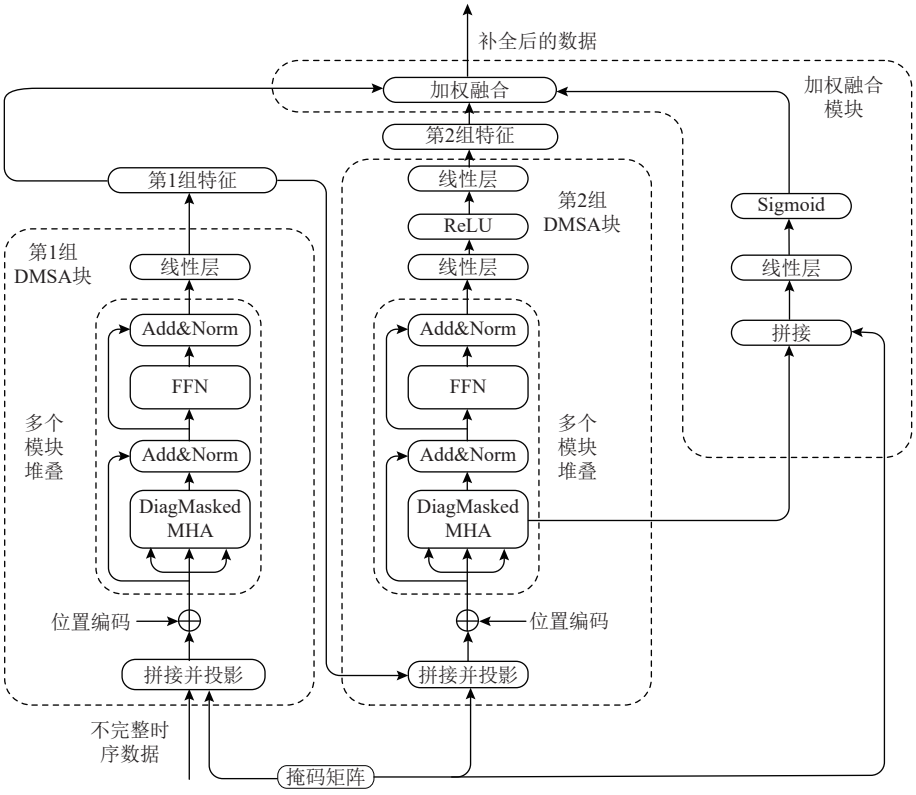


Fig. 8 Model architecture of SAITS[54]

图 8 SAITS 的模型结构[54]

塔结构，每一层节点均与同层相邻节点、父节点及子节点进行交互. Peri-midFormer 将金字塔中从顶层到底层的路径定义为周期表征流，通过聚合多条表征流输出最终的补全结果.

Nie 等人[56]提出的 ImputeFormer 在 Transformer 中引入了低秩结构先验，核心思想是通过低秩约束引导注意力机制在时空维度中聚焦于主要信号，抑制噪声. 如图 9 所示，该方法的时间注意力部分将原始序列压缩至低秩空间后再重建，有效降低计算复杂度，空间注意力部分则利用低维嵌入建构隐式的变量间相关性图. 为进一步提升补全质量，ImputeFormer 设计了傅里叶补全损失函数，利用时空序列在频域内的稀疏性约束补全结果的频谱能量集中，与低秩结构形成双重正则项. 整体训练采用随机掩码自监督方式，结合重构误差与傅里叶补全损失进行联合优化.

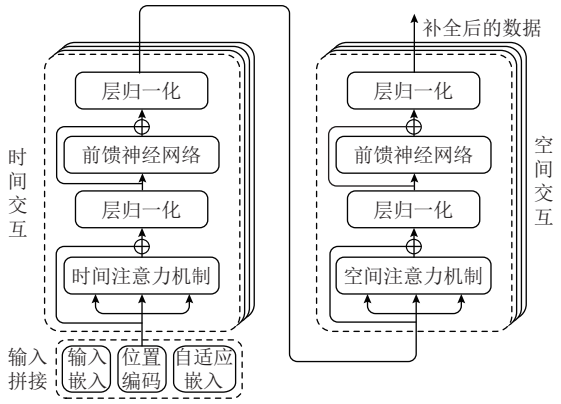


Fig. 9 Model architecture of ImputeFormer[56]

图 9 ImputeFormer 的模型结构[56]

Yang 等人[57]提出的 FGTI 将时间序列转化为频域信号，并通过扩散模型进行建模. 该方法使用高频滤波器和主导频率滤波器分别捕捉高频残差(短期波动)与主导频率(长期趋势). 在生成过程中引入双频注意力机制，时序注意力机制建模时序依赖，变量注意力机制建模变量间关系，两者共同成为条件引导模型生成补全分布.

Yuan 等人[58]提出的 Diffusion-TS 方法使用 Transformer 框架将时间序列显式地分解为趋势项、季节项与残差项 3 部分，使用多项式回归建模趋势项、傅里叶合成层(Fourier synthetic layers)建模季节项和残差项. 模型采用扩散模型逐步去噪，引入与 ImputeFormer 中类似的傅里叶损失，与重构误差一同约束模型生成符合真实分布的缺失值补全.

Zhang 等人[59]提出的 UP2ME 采用“单变量预训练-多变量微调”的策略. 该方法首先将任务数据集中的多变量时序拆解为单变量时序用于预训练，完全忽略变量间关系. 在具体任务中，UP2ME 冻结其 Transformer 的编码器和解码器参数，在编码器和解码器之间引入可学习的时序-变量层建模变量间关系和进一步的时序依赖. 该方法可适用于补全、预测、异常检测、分类等多种时序任务.

Wang 等人[60]提出的 TimeMixer++融合了 TCN 多尺度卷积和 TimesNet 多周期建模的思路. 图 10 示意了该方法的模型结构，TimeMixer++首先使用卷积层对时序数据进行下采样得到多尺度的时序片段，再使用快速傅里叶变换识别时序中的周期并将这些片段折叠为多分辨率的二维时序图像. 随后该方法通过双轴注意力机制捕捉周期内外的季节项与趋势项. 在对不同尺度不同分辨率的信息进行融合后，模型最终输出补全后的数据.

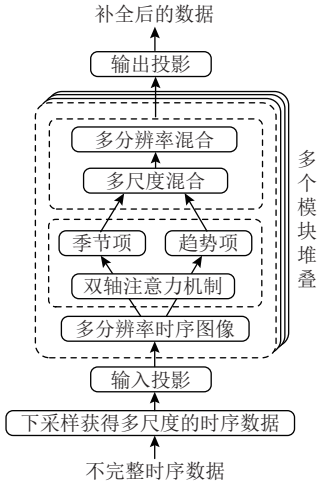


Fig. 10 Model architecture of TimeMixer++[60]

图 10 TimeMixer++的模型结构[60]

4.4 基于大模型的方法

Zhou 等人[61]提出的 GPT4TS 是较早将大模型引入时间序列分析的尝试，通过跨模态迁移学习，将预训练的自然语言处理(NLP)或计算机视觉(CV)模型(如 GPT-2[80]、BERT[81]、BEiT[82]等)迁移至时序任务中. 如图 11 所示，GPT4TS 冻结预训练大模型超过 90% 的参数，仅微调输入嵌入(input embedding)、位置编码和层归一化(layer normalization)，从而在保留原模型通用建模能力的同时适应时序特性. 模型的输出层则根据具体任务(如补全、预测等)进行灵活适配，实现对各种时序任务的迁移泛化.

Chen 等人[62]提出的 LLM-TS Integrator 方法利用大语言模型(如 LLaMA-3b[72])辅助优化传统时序模

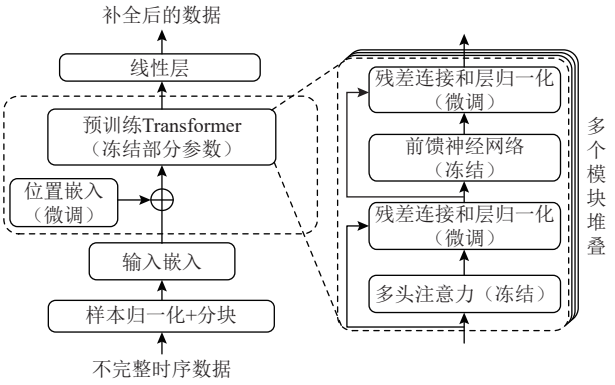


Fig. 11 Model architecture and training strategy of GPT4TS[61]

图 11 GPT4TS 的模型结构和训练策略[61]

型(如 TimesNet). 其框架包括互信息增强模块和样本重加权模块 2 部分. 前者通过最大化 TimesNet 生成的表征与大语言模型提供的文本描述之间的互信息, 使 TimesNet 在保留自身时序建模能力的前提下捕获全局信息. 后者根据预测损失动态调整样本对应的损失权重, 通过多层感知机学习其映射. 该方法使用双层优化策略(内层更新 TimesNet 模型参数, 外层更新权重分配网络)实现自适应训练. 训练完成后, TimesNet 在多类时序任务上均有明显提升.

Goswami 等人[63] 提出的 MOMENT 是一个大规模预训练的通用时间序列模型, 旨在应对跨领域、异质性强的时序数据建模需求. 他们构建了包含医疗、金融、工程等 5 个领域共计 20 万条序列的 Time Series Pile 数据集. 图 12 示意了 MOMENT 的模型结构和训练方式, 该模型采用掩码建模(masked modeling)任务, 通过 Transformer 编码器对约 30% 的随机遮蔽片段进行上下文重建学习, 使模型获得通用时序分析能力. 在补全任务中, MOMENT 可直接用于零样本补全或经微调后执行精确补全.

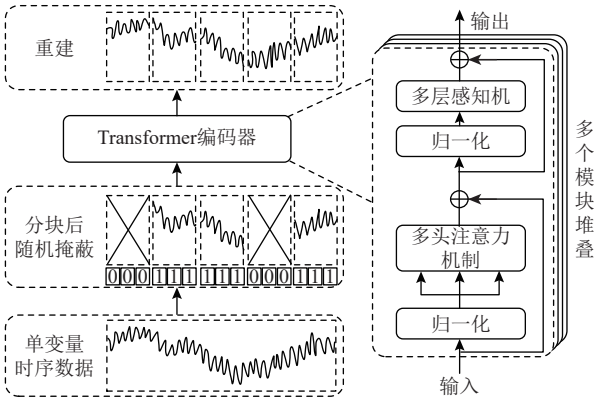


Fig. 12 Model architecture and training strategy of MOMENT[63]

图 12 MOMENT 的模型结构和训练策略[63]

Liu 等人[64] 提出的 Timer 是另一种大规模预训练基座模型, 构建了统一时间序列数据集 UTSD(涵盖 7 个领域), 将多变量数据转化为单变量数据池后使用滑动窗口生成训练片段. 不同于 MOMENT, Timer 采用解码器式 Transformer 架构, 通过不同的输出构造方式适配下游任务. 在补全任务中, Timer 基于上下文生成缺失部分并与观测值拼接.

Talukder 等人[65] 提出的 TOTEM 同样基于大规模预训练, 但不同于 MOMENT 使用连续表征建模, TOTEM 使用向量量化自编码器(VQVAE)将时间序列离散化为词元(token), 实现“类语言”的序列建模方式. 模型编码器采用一维卷积提取时序特征, 量化器将连续表征映射至预定义的词元空间, 解码器通过自注意力机制根据上下文信息重建缺失片段. 当缺失片段无法直接生成词元时, TOTEM 将根据上下文选择最合理的词元补全缺失内容, 增强了补全的结构连贯性与语义一致性.

4.5 优缺点分析

基于不同基础架构和增强技术的深度学习时序数据补全方法在建模能力、计算效率和泛化性能等方面各具优劣, 下文对其进行对比分析.

4.5.1 基础架构

基于 RNN 的方法天然适用于建模时间序列中的时序依赖. 一些观点认为 RNN 架构难以捕捉长期时序依赖, 将导致其输出质量下降[19], 改进版的 RNN(如 GRU)在一定程度上缓解了上述问题, 但其建模能力仍受限于网络深度和状态传播方式, 且其较低的计算效率限制了其应用.

基于 CNN 的方法有较强的局部依赖建模能力, 尤其适用于捕捉固定时间窗口内的时序周期依赖. CNN 方法因其良好的并行性具有很高的计算效率. 但受卷积核限制, CNN 缺乏显式的全局依赖建模机制, 在非周期性的时序数据上补全效果可能较弱.

基于注意力机制的方法可以实现全局依赖建模, 能很好地捕捉长期时序依赖和变量间关系, 并具有良好的并行性能. 然而, 注意力机制的计算复杂度随序列长度和变量数呈平方级增长, 在高维长序列上, 资源消耗和内存开销成为其主要瓶颈.

基于大模型的方法通过大规模预训练获取通用知识, 具备良好的跨模态跨任务的迁移能力, 可以在时序数据补全任务上实现零样本或少样本预测. 然而由于庞大的参数量, 大模型训练、微调、推理所需的资源较高. 此外, 部分大模型方法将多变量时序数据拆解为多条单变量时序数据进行处理, 弱化了对

变量间相关性的建模能力，有待进一步优化.

4.5.2 增强技术

GAN 能生成逼真的补全结果，但训练过程不稳定[15]，易发生模式崩溃，对超参数和初始化敏感. 扩散模型通过从噪声中逐步恢复缺失数据，能够生成更加平滑合理的补全分布，然而该类方法的训练与采样过程的资源开销大、收敛速度慢，在实际应用中受到限制. GNN 能很好地对具有明确空间关系的时序数据进行建模，然而在不具备这种空间信息的数据上，该类方法的表现十分依赖图结构构建过程的准确性. 通过快速傅里叶变换进行频域建模的方法能够挖掘时间序列中潜在周期与稀疏特性，这类方法的有效性与频谱分析的准确性有很大关系.

5 实验设计

为全面评估和选择合适的不完整时序数据补全方法，本文设计了一系列实验，评估在不同缺失场景、缺失率、变量数量、序列长度的时序数据上各方法的补全质量与效率，并进一步分析补全结果对下游任务的影响. 下文对实验所用数据集、评价指标、实验设置和缺失场景设计进行介绍.

5.1 数据集

表 4 对时序数据补全任务中的常用公开数据集进行整理，涵盖气象、电力、交通、能源及医疗等领域. 这些数据集在样本数量、变量数量、采样间隔及原始缺失率方面各有不同. 所有数据集均按时间顺序从前往后划分成测试集、验证集和训练集. 为量化数据集内各变量间的相关性，引入互信息（mutual information）[83]作为其度量指标，并将所有变量对的互信息取平均值作为整个数据集的变量相关性度量.

1）BeijingAir[4]. 包含 2013 年 3 月至 2017 年 2 月期间北京市 12 个空气质量监测站的每小时污染物观测数据，其中每个站点所返回的数据包含 11 个数值变量，将所有站点的变量聚合后，形成具有 132 个变量的数据集.

2）ItalyAir[84]. 记录 2004 年 3 月至 2005 年 2 月期间的意大利空气质量数据，包括金属氧化物和污染物在内的 13 个指标的每小时数值.

3）Electricity.由 2011 年 1 月至 2014 年 12 月期间 370 个用户的每小时的用电量构成. 部分用户在 2011 年之后才开始采集数据，在这之前其对应的数据被置为 0，因此这个数据集可被视为变量维度动态变化的时序数据集.

4）ETTm1[85]. 包含从 2016 年 7 月至 2018 年 7 月期间经过预处理的电力负荷和油温数据在内的 7 个变量，其采样间隔为 15 min.

5）ETTh1[85]. 与 ETTm1 数据集结构相似，但采样间隔为 1 h.

6）PeMS[3]. 记录旧金山湾区高速公路上 862 个传感器测量的每小时道路占有率数据.

7）Solar Alabama.包含美国阿拉巴马州各光伏电站于 2006 年的太阳能发电数据，其采样间隔为 10 min. 由于在夜晚时光伏电站不存在发电量，该数据集呈现明显的昼夜节律.

8）PhysioNet2012[7]. 包含约 12 000 条 ICU 患者记录，原始目标是利用入院后 48 h 内的监测数据预测患者院内死亡率. 由于 ICU 数据的特殊性，多数指标

Table 4 Commonly Used Open-source Datasets for Time Series Imputation Tasks

表 4 时序数据补全任务中常用的开源数据集

领域	数据集	样本数量	变量数量	序列长度	采样间隔	数据集划分（训练集，验证集，测试集）	原始缺失率/%	平均互信息
气象	BeijingAir[4]	1 458	132	24	1 h	（852，305，303）	1.6	0.312
	ItalyAir[84]	387	13	24	1 h	（233，77，77）	0	0.434
电力	Electricity①	1 460	370	96	1 h	（852，305，303）	0	0.379
	ETTm1[85]	730	7	96	15 min	（427，153，145）	0	0.352
	ETTh1[85]	730	7	24	1 h	（427，153，145）	0	0.352
交通	PeMS[3]	731	862	24	1 h	（456，122，153）	0	0.886
能源	Solar Alabama②	365	137	144	10 min	（183，91，91）	0	0.885
医疗	PhysioNet2012[7]	11 988	37	48	1 h	（2 557，640，800）	79.7	0.038
	PhysioNet2019[86]	6 104	34	48	1 h	（3 152，789，986）	79.6	0.040

① https://archive.ics.uci.edu/dataset/321/electricityloaddiagrams20112014

② https://www.nrel.gov/grid/solar-power-data.html

不会被完整记录,该数据集的原始缺失率为 80%.

9) PhysioNet2019[86]. 收录 3 家医院共 40 336 条 ICU 住院数据,每条记录包含 40 个临床和生理变量,该数据集的原始缺失率亦达到约 80%.

5.2 评价指标

在不完整时序数据补全任务中,通常使用补全后的数值与真实数值的差异衡量方法的补全质量,常用的评价指标有平均绝对误差(mean absolute error, *MAE*)和均方根误差(root mean squared error, *RMSE*). 当需要消除量纲带来的影响时,可采用平均相对误差(mean relative error, *MRE*). *MAE*、*RMSE*、*MRE* 指标在时序补全任务中的定义为:

$$MAE(\hat{\boldsymbol{X}},\boldsymbol{X},\boldsymbol{M})=\frac{\sum_{j=1}^{d}\sum_{i=1}^{n}(1-m_i^j)\times\left|\hat{x}_i^j-x_i^j\right|}{\sum_{j=1}^{d}\sum_{i=1}^{n}(1-m_i^j)}, \tag{17}$$

$$RMSE(\hat{\boldsymbol{X}},\boldsymbol{X},\boldsymbol{M})=\sqrt{\frac{\sum_{j=1}^{d}\sum_{i=1}^{n}(1-m_i^j)\times(\hat{x}_i^j-x_i^j)^2}{\sum_{j=1}^{d}\sum_{i=1}^{n}(1-m_i^j)}}, \tag{18}$$

$$MRE(\hat{\boldsymbol{X}},\boldsymbol{X},\boldsymbol{M})=\frac{\sum_{j=1}^{d}\sum_{i=1}^{n}(1-m_i^j)\times\left|\hat{x}_i^j-x_i^j\right|}{\sum_{j=1}^{d}\sum_{i=1}^{n}(1-m_i^j)\times\left|x_i^j\right|}. \tag{19}$$

5.3 实验设置

本文在上述 9 个数据集上对 2 种统计方法(Mean, LOCF[31])、2 种机器学习方法(SoftImpute[39], TRMF[40])和 10 种深度学习方法(BRITS[42], SSGAN[46], TimesNet[48], ModernTCN[49], SAITS[54], ImputeFormer[56], TimeMixer++[60], GPT4TS[61], MOMENT[63], TOTEM[63])进行了系统实验. 评估内容涵盖缺失场景、缺失率、变量数量、序列长度等对方法补全质量和效率的影响. 同时引入下游分类与预测任务以分析补全结果对后续任务的提升作用.

所有实验均在统一的实验环境下进行: Intel® Xeon® Gold 6326 CPU @ 2.90 GHz, 256 GB 内存, NVIDIA L40(48 GB)显卡, Ubuntu 20.04.6 系统. 实验基于 PyTorch 和 Pypots[87] 框架构建,并根据实验需求进行调整. 为降低随机性干扰,每项实验均重复运行 5 次并取平均值作为实验结果. 对于深度学习方法,每次实验的训练 Epoch 数均为 100,若验证集上的损失在连续 10 个 Epoch 内未降低,则提前停止训练,并保留验证集损失最小的模型用于补全任务. 模型优化过程中所采用的优化器为 Adam[88] 并设置学习率为 1×10^{-3},其余超参数遵循各方法原始论文或官方补全任务代码的默认设置. 需要说明的是,超参数可在一定程度上影响深度学习模型性能,在不同数据集上进行专门调参可进一步提升补全质量,而本文在所有数据集上使用了相同的默认设置,故深度学习方法在具体数据集上的表现仍有提升空间.

一些方法(如 TimesNet, GPT4TS 等)的原始训练过程中直接将缺失位置对应的真实值作为标签,这与实际应用中无法获知缺失位置真实值的情形不符. 为更贴合真实场景,本文借鉴 SAITS[54] 的训练策略,在存在缺失值的训练集上再人为遮蔽部分观测值,构造补全目标作为模型的训练标签.

5.4 缺失场景设计

本文设计的缺失场景是结合时序数据的特有缺失模式[19] 与 Rubin[30] 的缺失机制理论进行设计得到的. 本文将时序数据的缺失模式概括为 3 类:

1) 点状缺失(point missing). 指某一变量在某一时间步上缺失,是最基础的缺失模式.

2) 块状缺失(block missing). 指某一变量在一段连续时间内缺失,是沿时间维度的连续缺失.

3) 全变量缺失(blackout). 指部分时间步上所有变量同时缺失,是沿变量维度的连续缺失.

在此基础上,本文结合 Rubin 的缺失机制理论中的 3 种缺失机制 MCAR, MAR 和 MNAR,设计实验中的缺失场景. 首先考虑点状缺失模式下结合缺失机制的场景:

1) MCAR-point. 点状缺失值的位置完全随机.

2) MAR-point. 点状缺失值的位置与其余观测值相关. 在实验中,使用其余变量对当前变量是否缺失的概率建立逻辑回归模型[89] 来模拟这一场景.

3) MNAR-point. 点状缺失值的位置与其自身有关. 在实验中,变量的缺失概率与其所在时间步有关,采用指数分布模拟越早被记录的数据越容易发生缺失的情形.

上述场景设计可进一步扩展至 block missing 和 blackout 两种模式上,最终构建出 9 种典型的缺失场景. block missing 模式下的连续缺失时间块大小为 2～24 之间的随机数, blackout 模式下某些时间步上的所有变量均缺失. 这种设计可以模拟现实世界中的时序数据缺失场景,例如 MAR-block 可被视为在某些极端天气环境下,一部分传感器数据缺失概率

更大的场景，MCAR-blackout-block 则可被视为整个区域停电导致所有传感器在一段时间内的数据全部丢失的场景. 图 13 展示了在 ETTm1 数据集上模拟不同缺失场景时缺失值位置的分布情况.

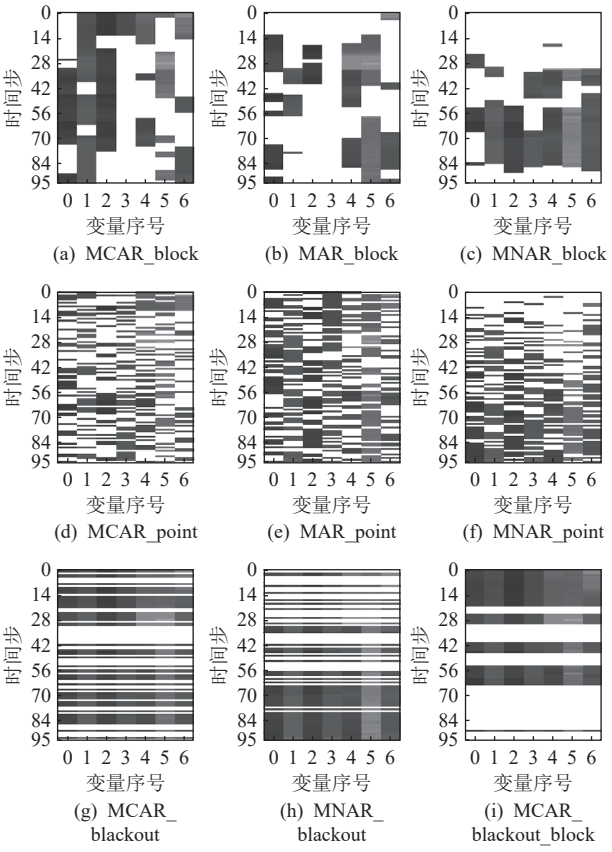
6 实验结果及分析

本节对设计的各实验的结果进行分析，随后总结实验结果，最后结合数据集特征推荐合适的方法.

6.1 缺失场景

实验在 ETTm1 和 Solar Alabama 两个变量相关性差异较大的数据集上进行. 图 14 分别为这 2 个数据集的互信息矩阵，ETTm1 数据集中变量之间的相关性较弱，而 Solar Alabama 数据集中变量之间的相关性较强(位于同一地区的多个太阳能发电站的发电量具有相似的变化趋势). 实验在每个数据集上模拟了 9 种不同的缺失场景，置入的缺失值比例固定在 50%.

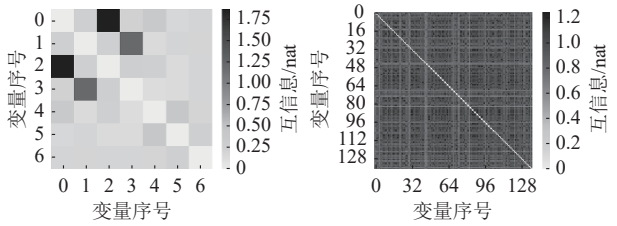
表 5 展示了各类方法在不同缺失场景下的补全



注：灰色区域为可观测位置，白色区域为缺失位置，每个子图展示了 ETTm1 数据集前 96 个时间步中的缺失位置.

Fig. 13 Nine simulated missing scenarios on ETTm1 dataset

图 13 在 ETTm1 数据集上模拟的 9 种缺失场景



注：灰色越深表示变量间相关性越强.

Fig. 14 Mutual information matrices of ETTm1 and Solar Alabama datasets

图 14 ETTm1 和 Solar Alabama 数据集的互信息矩阵

误差. 从总体趋势上看，在缺失模式相同的情况下(同为 point missing、block missing 或 blackout)，MNAR 缺失机制下的补全结果普遍较差，MAR 次之，MCAR 最好. 在缺失模式方面，point missing 模式由于缺失值离散分布，相对更易补全，而 block missing 与 blackout 模式因存在时间维度或变量维度上的连续缺失，补全难度明显增大.

对比 2 个数据集的实验结果可发现，变量相关性对补全方法在各场景下的补全误差存在影响. 在 ETTm1(低相关性)数据集上，多数方法在 blackout 模式下的补全结果优于 block missing 模式，而在 Solar Alabama(高相关性)数据集上则相反. 可能的原因是当数据集中的变量相关性较弱时，补全方法难以从其余观测变量学习到对缺失值补全有益的信息，而 block missing 模式下沿时间维度的信息缺失严重，两者叠加导致各方法的补全质量下降.

统计方法中，Mean 方法在所有场景下的补全质量较差. LOCF 方法在 point missing 和 blackout 模式下的补全质量可与部分深度学习方法相当，但在 block missing 模式下由于其依赖于历史观测值的局限性，补全质量明显降低.

机器学习方法 SoftImpute 和 TRMF 在 ETTm1 数据集上整体效果不理想，但在 Solar Alabama 数据集上的部分缺失场景下中表现突出，尤其是 TRMF 在 MCAR-block、MAR-block 及 MAR-point 上的补全质量要超过深度学习方法. 然而 TRMF 在 blackout 模式下的性能明显下降. 这类机器学习方法可能更适用于变量相关性高，且不存在全变量缺失的情况.

深度学习方法中，不同架构的方法在不同场景下展现出各自的特点. 基于注意力机制的方法在 ETTm1 数据集上的多数场景下取得最优结果，显示出其在低变量相关性条件下对时序依赖的建模能力. 而基于 RNN 的方法在 Solar Alabama 数据集上表现

Table 5 Imputation *RMSE* Under Different Missing Scenarios

表 5 不同缺失场景下的补全误差 *RMSE*

方法分类	具体方法	ETTm1 数据集								
		MCAR-block	MAR-block	MNAR-block	MCAR-point	MAR-point	MNAR-point	MCAR-blackout	MNAR-blackout	MCAR-blackout-block
统计方法	Mean	0.987±0.009	1.121±0.061	1.105±0.006	0.987±0.002	1.118±0.063	1.009±0.002	0.986±0.005	1.019±0.005	0.988±0.009
	LOCF	0.945±0.022	1.035±0.101	0.996±0.009	0.374±0.002	0.482±0.053	0.556±0.007	0.369±0.007	0.465±0.010	0.946±0.037
机器学习	SoftImpute	0.924±0.021	1.001±0.105	0.966±0.005	0.940±0.007	1.004±0.114	0.937±0.005	1.116±0.003	1.153±0.004	1.122±0.020
	TRMF	0.742±0.018	0.759±0.054	0.911±0.008	0.746±0.007	0.785±0.110	0.797±0.007	1.063±0.004	1.096±0.004	1.067±0.020
基于 RNN	BRITS*	0.654±0.024	0.725±0.075	0.818±0.018	0.263±0.005	**0.352±0.062**	0.344±0.009	0.313±0.005	0.362±0.006	0.764±0.029
	SSGAN*	0.521±0.020	0.654±0.056	0.810±0.012	0.265±0.005	0.369±0.069	0.372±0.009	0.349±0.008	0.381±0.014	0.796±0.020
基于 CNN	TimesNet	0.886±0.020	0.952±0.094	0.936±0.024	0.469±0.022	0.687±0.114	0.573±0.014	0.484±0.025	0.564±0.017	0.973±0.019
	ModernTCN	0.948±0.038	1.028±0.079	1.029±0.009	0.444±0.042	0.767±0.148	0.544±0.015	0.455±0.028	0.498±0.014	0.936±0.030
基于注意力机制	SAITS*	**0.476±0.050**	0.641±0.102	**0.704±0.045**	**0.241±0.009**	0.412±0.148	**0.258±0.004**	**0.303±0.014**	**0.329±0.008**	**0.562±0.034**
	ImputeFormer*	0.480±0.012	**0.637±0.055**	0.760±0.021	0.321±0.004	0.381±0.039	0.389±0.091	0.407±0.026	0.495±0.100	0.570±0.051
	TimeMixer++	0.630±0.030	0.740±0.074	0.824±0.022	0.462±0.007	0.614±0.079	0.576±0.049	0.471±0.008	0.549±0.020	0.648±0.033
基于大模型	GPT4TS	0.804±0.040		0.861±0.038	0.273±0.005	0.468±0.093	0.380±0.012	0.345±0.008	0.429±0.009	0.904±0.026
	MOMENT	1.148±0.025	1.129±0.127	1.107±0.062	0.926±0.128	1.090±0.177	1.024±0.161	1.043±0.130	1.095±0.160	1.156±0.038
	TOTEM	1.089±0.047	1.153±0.079	1.101±0.005	1.032±0.183	1.157±0.081	1.124±0.005	1.082±0.059	1.134±0.048	1.125±0.013

方法分类	具体方法	Solar Alabama 数据集								
		MCAR-block	MAR-block	MNAR-block	MCAR-point	MAR-point	MNAR-point	MCAR-blackout	MNAR-blackout	MCAR-block-blackout
统计方法	Mean	0.864±0.003	1.083±0.009	0.961±0.001	0.863±0.000	0.980±0.013	0.863±0.000	0.864±0.004	0.816±0.004	0.871±0.028
	LOCF	0.789±0.001	0.878±0.012	0.770±0.001	0.267±0.001	0.404±0.012	0.559±0.001	0.266±0.008	0.418±0.016	0.797±0.030
机器学习	SoftImpute	0.203±0.001	0.221±0.007	0.714±0.002	0.202±0.000	0.207±0.004	0.331±0.002	0.879±0.002	0.840±0.003	0.888±0.011
	TRMF	**0.181±0.002**	**0.195±0.005**	0.645±0.004	0.174±0.001	**0.175±0.004**	0.275±0.003	0.851±0.003	0.805±0.004	0.866±0.011
基于 RNN	BRITS*	0.187±0.001	0.206±0.003	0.520±0.023	0.181±0.000	0.186±0.006	0.251±0.009	**0.221±0.004**	**0.265±0.008**	0.721±0.026
	SSGAN*	0.188±0.001	0.242±0.006	0.709±0.021	0.167±0.001	0.207±0.009	0.382±0.011	0.470±0.011	0.512±0.022	0.755±0.022
基于 CNN	TimesNet	0.256±0.011	0.294±0.026	0.587±0.007	0.234±0.015	0.248±0.024	**0.235±0.011**	0.320±0.006	0.408±0.016	0.768±0.022
	ModernTCN	0.546±0.034	0.597±0.036	0.716±0.014	0.375±0.047	0.405±0.038	0.444±0.020	0.339±0.008	0.389±0.009	0.726±0.031
基于注意力机制	SAITS*	0.237±0.027	0.221±0.012	**0.455±0.074**	0.247±0.027	0.209±0.007	0.267±0.010	0.854±0.031	0.835±0.014	0.890±0.023
	ImputeFormer*	0.271±0.006	0.297±0.023	0.689±0.029	**0.145±0.013**	0.228±0.047	0.269±0.018	0.882±0.004	0.833±0.009	0.882±0.018
	TimeMixer++	0.377±0.003	0.436±0.022	1.069±0.044	0.371±0.003	0.384±0.009	0.628±0.012	0.407±0.003	0.560±0.030	**0.673±0.015**
基于大模型	GPT4TS				0.205±0.004	0.521±0.021	0.412±0.059	0.621±0.039	0.842±0.012	
	MOMENT	0.599±0.202	0.766±0.266	0.974±0.048	0.710±0.418	0.634±0.306	0.840±0.181	0.687±0.268	0.631±0.166	0.867±0.087
	TOTEM	0.871±0.007	0.894±0.028	0.941±0.013	0.730±0.181	0.830±0.081	0.841±0.058	0.739±0.198	0.833±0.024	0.877±0.031

注：黑体的数值表示当前缺失场景下最小的补全误差，下划线的数值表示次小的补全误差，专用于补全的深度学习方法用*标记.

更佳,这类方法可能在学习变量依赖方面更具优势.基于 CNN 的方法理论上适合局部时序依赖建模,实验中也体现出其在 block missing 模式下性能下降较为明显,表明其面对长时间连续缺失的短板.

专门设计用于补全任务的方法 BRITS、SSGAN、SAITS 和 ImputeFormer 的整体表现较优,这些方法相对于其他通用时序分析方法在多数缺失场景下具备更好的补全性能与鲁棒性.然而也存在例外,如在 ETTm1 数据集上,BRITS 和 SSGAN 等 RNN 方法,在面对长时间连续缺失时易出现误差累计,导致其在 block missing 模式下补全质量明显下降;而在 Solar Alabama 数据集下,SAITS 和 ImputeFormer 在 blackout 模式下补全误差较高,由于所有变量在部分时间步上全部缺失,缺乏变量间的关联信息使基于注意力机制的补全变得困难.通用时序分析方法中,TimesNet 和 TimeMixer++表现出相对较好的性能,但整体仍不如专为补全设计的深度学习方法.

大模型方法相对于传统深度学习方法并没有展

现出优势，GPT4TS 在 point missing 模式下表现优秀，但在部分缺失位置密集的场景中，其生成的补全结果仍包含缺失值，导致训练过程中无法正常进行梯度计算从而无法正常运行. MOMENT 和 TOTEM 在大多数场景下补全误差较高且不稳定，原因可能在于其原始设计中对补全任务的设置与本文的实验设计存在差异（如 MOMENT 将所有具有缺失值的时序片段视为全部缺失片段并进行遮蔽，导致其无法处理实验中每个时序片段都存在缺失的情况）.

6.2 缺失率

本文在 8 个数据集的观测值中置入了缺失率为 10%~90%、缺失场景为 MCAR-point 的缺失值，以评估不同缺失率下各方法的补全质量与运行时间变化趋势. 实验中除 PhysioNet2019 数据集的缺失率控制在 81%~89% 外，其余数据集的缺失率均为 10%~90%.

图 15 展示了各方法在不同缺失率下的补全误差 *RMSE*，*RMSE* 越小表示补全质量越高. 除 Mean 方法外，其余方法的补全误差随缺失率的增加而上升. 统计方法 LOCF 在低缺失率下表现良好，但其补全误差随缺失率增加迅速上升. 机器学习方法变量相关性较高的数据集上补全质量较高，如 PeMS 和 Solar Alabama 数据集，再次验证了缺失场景实验的分析.

深度学习方法在大多数变量相关性较低的数据集上优于统计方法与机器学习方法，体现了其在建模复杂时序依赖与非线性结构方面的优势. 但在 Electricity 数据集上，补全性能反而不如部分传统方法，该数据集中变量维度随时间动态变化，不符合多数深度学习方法对固定输入维度的假设，而统计方法、机器学习方法及 GPT4TS（将多变量时序数据拆解为多个单变量时序数据进行处理）则能更好地适应这种情形. 与在缺失场景实验中得出的结果类似，BRITS、SSGAN、SAITS 和 ImputeFormer 等专用于时序数据补全的方法分别在一些数据集上有最佳表现. GPT4TS 在低缺失率情况下表现优秀，但如前文所述，过高的缺失率将导致该方法无法正常运行. 在通用时序分析方法中，TimesNet 和 TimeMixer++的补全质量相对较好.

图 16 展示了各方法在不同数据集上的平均运行时间. Electricity 数据集包含最多的观测点，各方法在该数据集上的运行时间较长. 基于 RNN 的方法 BRITS 和 SSGAN 在所有数据集上的平均运行时间较长，这源于其自回归建模方式带来的低并行性. MOMENT 的运行时间亦较长，主要由于其较大的模型参数导致微调耗时较长. ImputeFormer 在变量维度较高的数据集中运行时间偏长. 相对而言，基于 CNN 以及基于注意力机制的方法（GPT4TS 和 TOTEM 也是基于注意力机制的大模型方法）因其良好的并行计算特性，运行效率较高.

6.3 变量数量

本文在 Electricity 和 PeMS 两个数据集上通过调整时序数据中的变量数量，以评估变量数量对深度学习方法补全质量与效率的影响. 每个数据集中随机置入缺失率为 50%、缺失场景为 MCAR-point 的缺失值，并从数据集中随机选择变量数量分别为 1, 2, 4, 8, 16, 32, 64, 128, 256, 370, 512 和 862 的子集进行实验. 考虑到不同变量组合可能存在的量纲差异，采用 *MRE* 指标作为衡量补全质量的标准.

如图 17 所示，随着变量数量的增加，多数方法的补全误差 *MRE* 呈现先下降后上升的趋势，并在较大变量数量下趋于平稳. 在 Electricity 数据集上，当变量数量为 8 时，各方法有最优补全质量；而在 PeMS 数据集上这个数量为 32. 该现象可能归因于 PeMS 数据集中变量之间相关性更强，额外引入的变量为模型提供了更多上下文信息，更有利于对缺失值的估计. 总体而言，相较于单一变量，多变量输入提升了深度学习方法的补全质量，说明模型能够有效从其余变量中提取对目标变量有益的补全信息，但随着变量数量进一步增多（如超过数百个），模型在集成多变量信息的能力趋于饱和，甚至可能受到冗余信息干扰，导致边际增益减弱.

如图 18 所示，在运行时间方面，除了基于 RNN 的方法 BRITS 和 SSGAN 外，其余方法的运行时间随着变量数量的增加呈上升趋势. RNN 方法在时间维度上采用自回归结构，导致低并行度，但在变量维度上可并行处理多个输入变量，因此对变量数量的变化相对不敏感. 然而，RNN 方法的总体计算效率在变量数量不太多（如 256 或以下）时仍低于其他并行性良好的方法. 反之，在一个变量数量达到几千甚至上万的数据集上，随着其他方法的计算复杂度上升，RNN 方法的计算效率可能不再成为其劣势.

6.4 序列长度

本文在 BeijingAir 和 ETTm1 两个数据集上通过调整输入序列长度，以评估序列长度对深度学习方法补全质量与效率的影响. 每个数据集中随机置入缺失率为 50%、缺失场景为 MCAR-point 的缺失值，输入的序列长度分别被设置为 24, 48, 96, 192, 384, 768, 1 536 个时间步.

如图 19 所示，多数方法在输入序列过短或者过

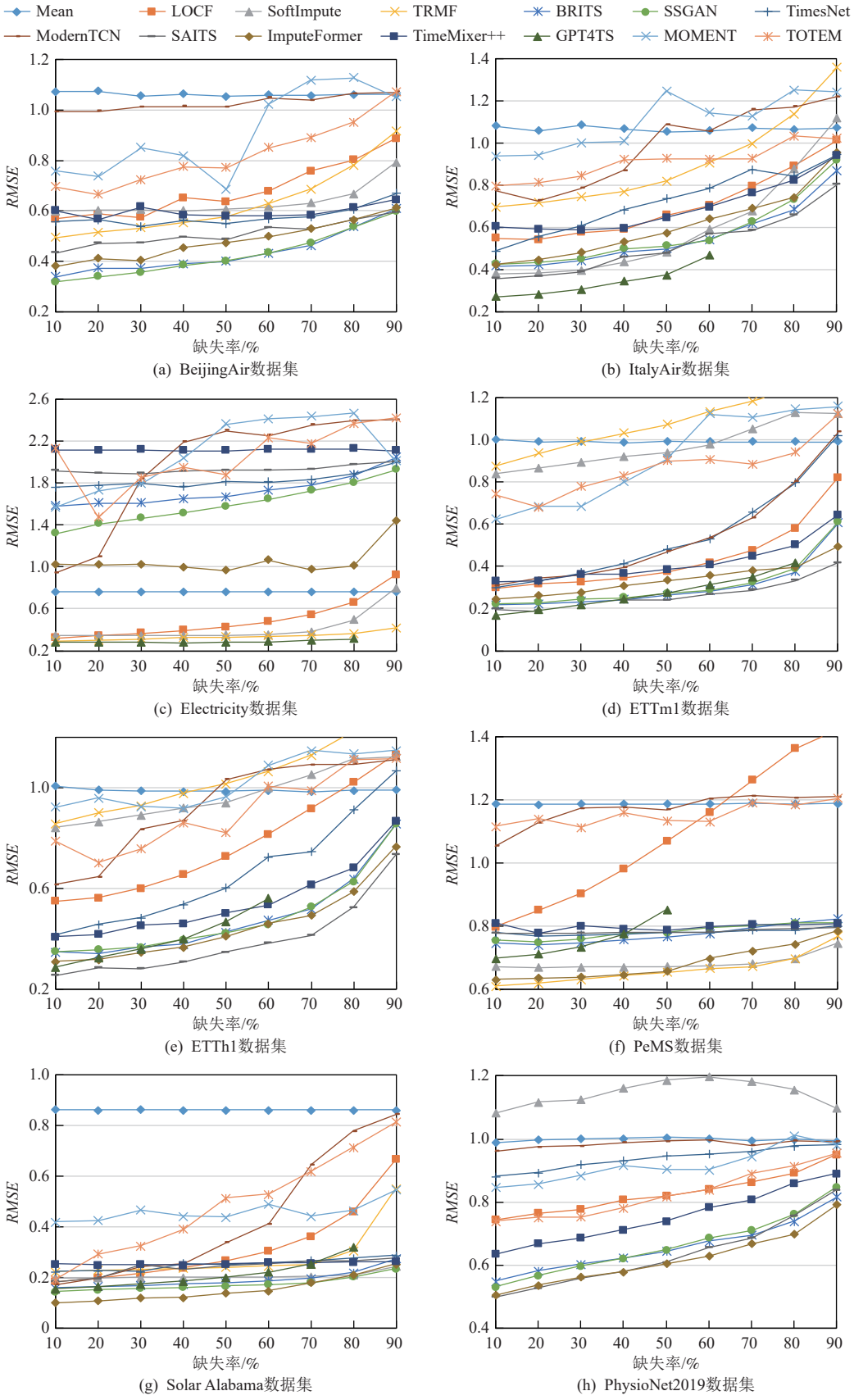
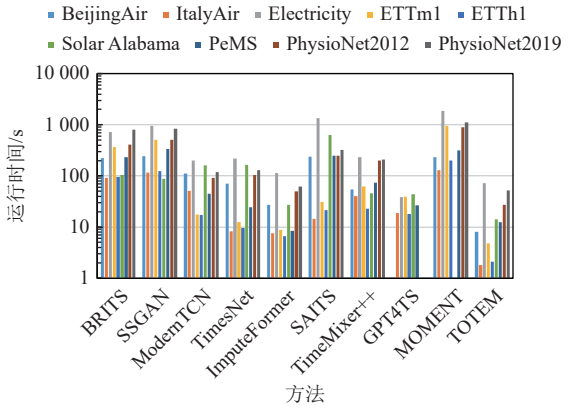


Fig. 15 Imputation *RMSE* of different methods under varying missing rates on time series datasets

图 15 补全方法在各个时序数据集上不同缺失率下的补全误差 *RMSE*



注：时间坐标轴为对数刻度.

Fig. 16 Runtime of imputation methods across datasets

图 16 补全方法在各数据集上的运行时间

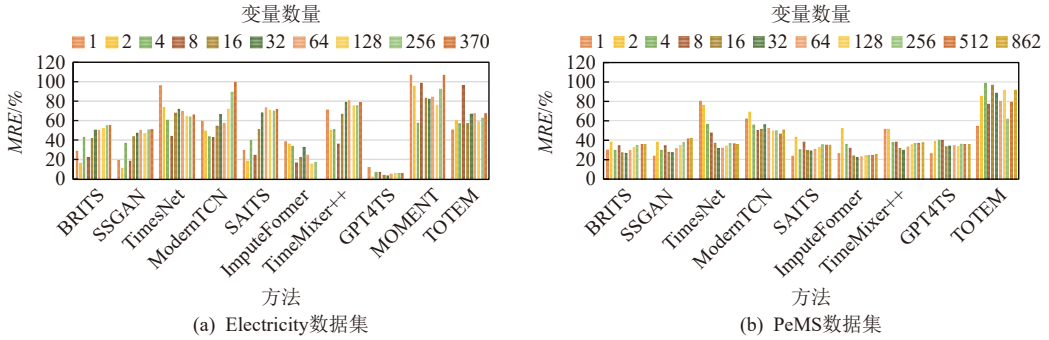
长时补全误差都会有所上升. 输入序列过短时, 补全方法可能无法捕捉长期依赖, 而输入序列过长时补全方法可能无法聚焦关键时序片段, 甚至因梯度爆炸、注意力稀释等问题影响补全质量. ImputeFormer 和 MOMENT 方法在输入序列长度过长时可能会由于显存不足而无法正常运行. 尽管理论上 RNN 方法较难捕捉长时序依赖, 但实验结果表明 BRITS 和 SSGAN 的补全质量随着输入序列长度的增加维持在一个良好的水平. 这可能归因于目标数据集的补全任务不依赖于长期时序数据, 而更依赖临近观测.

如图 20 展示了各方法在不同输入序列长度下的运行时间. RNN 方法 BRITS 和 SSGAN 的运行时间与输入序列长度呈正相关, 而其余方法的运行时间随输入序列长度的增长呈下降趋势. 这主要是由于输入序列长度增加的同时, 训练时序样本段数减少, 非 RNN 方法更易并行处理长序列, 因此其整体运行效率反而提高了.

6.5 下游任务

时序补全任务的主要目标之一是为后续的时序分析任务提供更可靠的数据基础. 因此, 将补全后的时序数据应用于具体的下游任务, 是评估补全方法实际有效性的重要环节. 本实验通过分类和预测 2 个典型任务验证不同补全方法对下游任务的提升作用.

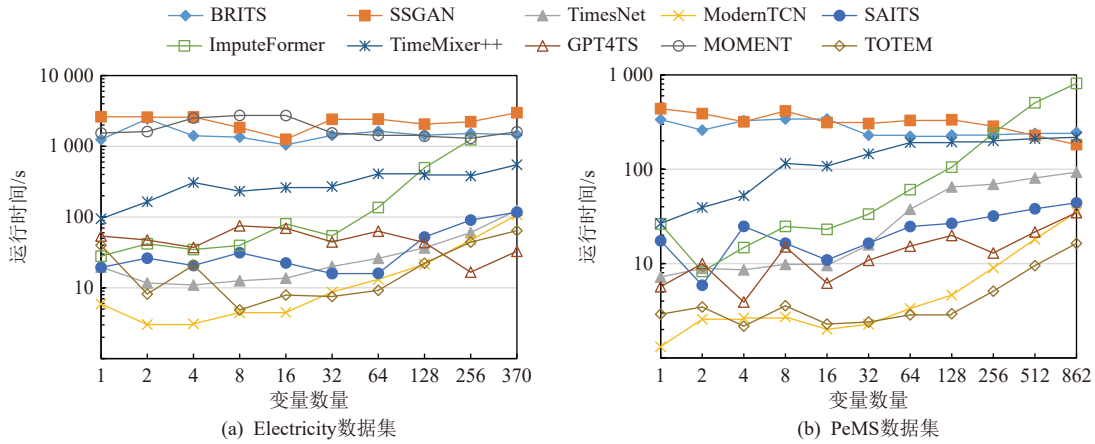
下游分类任务在 PhysioNet2012 数据集上进行. 实验首先使用各类补全方法分别对原始数据集进行



(a) Electricity数据集 (b) PeMS数据集

Fig. 17 Imputation *MRE* across different variable numbers

图 17 不同变量数量下的补全误差 *MRE*

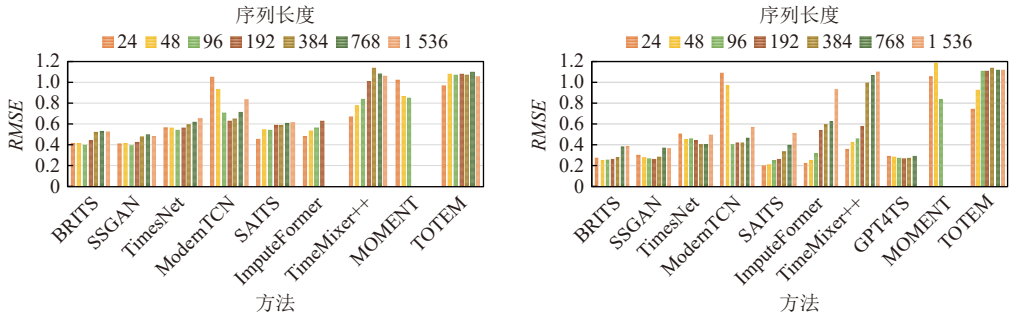


(a) Electricity数据集 (b) PeMS数据集

注：时间坐标轴为对数刻度.

Fig. 18 Impact of different numbers of variables on runtime

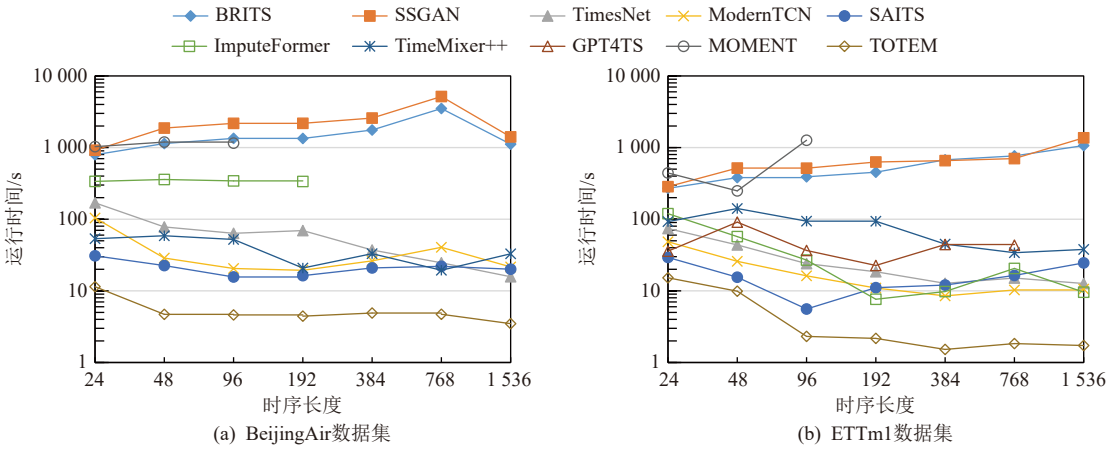
图 18 不同变量数量对运行时间的影响



(a) BeijingAir数据集 (b) ETTm1数据集

Fig. 19 Imputation *RMSE* across different sequence lengths

图 19 不同序列长度下的补全误差 *RMSE*



(a) BeijingAir数据集 (b) ETTm1数据集

注：时间坐标轴为对数刻度.

Fig. 20 Impact of different sequence lengths on runtime

图 20 不同序列长度对运行时间的影响

补全, 随后使用补全后的训练集分别对一个 GRU-D[41] 时序分类模型进行训练, 将经过训练的模型用于预测 PhysioNet2012 测试集中的标签(进入 ICU 48 小时内患者是否死亡). 在这个分类任务中, 采用 ROC_AUC(receiver operating characteristic area undercurve)和 PR_AUC(precision-recall area under curve)指标衡量分类结果的质量, 2 项指标数值越高表示模型的分类准确度越高. 实验中, 所有 GRU-D 分类模型采用相同的超参数设置, 唯一的不同在于训练数据来自不同补全方法的补全结果. 如表 6 所示, 经过补全的数据的确提高了下游模型的分类能力. 相比于统计方法与机器学习方法中较优的 LOCF 方法, 经过深度学习方法 TOTEM 补全的数据使得下游分类任务的指标数值提升了约 5%. 值得注意的是, TOTEM 并不是在 PhysioNet2012 数据集中补全误差最低的方法, 但其补全结果所训练出的 GRU-D 分类器具有最优的分类性能, 可见单纯使用 *RMSE* 等数值指标评价补全方法性能的做法并不全面.

下游预测任务在 ETTm1 和 Solar Alabama 数据集

Table 6 Downstream Classification Accuracy on PhysioNet2012 Dataset

表 6 下游 PhysioNet2012 数据集分类任务的准确性

方法分类	具体方法	ROC_AUC↑	PR_AUC↑
统计方法	Mean	0.705±0.035	0.417±0.054
	LOCF	0.703±0.014	0.450±0.019
机器学习	SoftImpute	0.665±0.069	0.345±0.169
	TRMF	0.627±0.057	0.396±0.218
基于 RNN	BRITS*	0.705±0.013	**0.502±0.073**
	SSGAN*	0.713±0.031	0.457±0.023
基于 CNN	TimesNet	0.702±0.022	0.447±0.041
	ModernTCN	0.668±0.031	0.357±0.051
基于注意力机制	SAITS*	0.704±0.005	0.451±0.027
	ImputeFormer*	<u>0.730±0.026</u>	0.476±0.012
	TimeMixer++	0.710±0.026	0.431±0.024
基于大模型	MOMENT	0.669±0.015	0.449±0.035
	TOTEM	**0.743±0.018**	<u>0.496±0.030</u>

注：黑体的数值表示最优值；下划线的数值表示次优值；专用于补全的深度学习方法用*标记；“↑”表示越大越好.

上进行. 首先在数据集中随机置入缺失率为 50%、缺失场景为 MCAR-point 的缺失值, 再使用补全方法对这些不完整数据进行补全, 随后使用补全后的数据训练一个 Transformer[14] 时序数据预测模型并预测输入时序之后 24 步的时间序列, 通过 *RMSE* 衡量预测结果的质量. 如表 7 所示, 在 ETTm1 数据集上具有最佳预测结果的模型是由 ModernTCN 和 ImputeFormer 的补全数据训练的, 而在 Solar Alabama 数据集上表现最优的模型由 BRITS 和 SAITS 的补全数据训练得到. 与分类任务不同, 补全误差较低的方法往往能带来更优的下游预测表现, 补全误差(如 *RMSE*)与下游预测误差之间具有更高的一致性.

6.6 实验总结

本文系统评估了多种时序数据补全方法在不同实验设计下的表现, 实验得到的主要结论如下:

1) 补全方法的补全质量与效率受缺失场景、缺失率、变量数量、序列长度及变量间相关性等因素影响. 深度学习方法并非在所有场景下都是最优选择. 若数据缺失率较低且对补全准确度要求不高, 统计方法如 LOCF 可作为快速补全手段. 在变量间相关性较高、无全变量缺失的场景中, SoftImpute 与 TRMF 等机器学习方法在效率与准确度之间取得良好平衡. 而在追求最佳补全质量的情形下, 深度学习方法往往是理想的选择.

Table 7 Prediction *RMSE* of the Downstream Prediction Tasks

表 7 下游预测任务的预测误差 *RMSE*

方法分类	具体方法	ETTm1	Solar Alabama
统计方法	Mean	0.541±0.022	0.738±0.060
	LOCF	0.508±0.008	0.681±0.177
机器学习	SoftImpute	0.660±0.066	0.590±0.251
	TRMF	0.567±0.026	0.603±0.119
基于 RNN	BRITS*	0.512±0.023	**0.540±0.131**
	SSGAN*	0.498±0.015	0.680±0.322
基于 CNN	TimesNet	0.514±0.009	0.617±0.219
	ModernTCN	**0.484±0.016**	0.712±0.157
基于注意力机制	SAITS*	0.500±0.013	<u>0.570±0.182</u>
	ImputeFormer*	<u>0.491±0.008</u>	0.893±0.378
	TimeMixer++	0.500±0.020	0.748±0.208
基于大模型	GPT4TS	0.511±0.016	0.739±0.154
	MOMENT	0.576±0.020	0.712±0.285
	TOTEM	0.552±0.014	0.635±0.111

注: 黑体的数值表示最优值; 下划线的数值表示次优值; 专用于补全的深度学习方法用*标记.

2) 专门设计用于时序数据补全的方法 BRITS、SSGAN、SAITS 和 ImputeFormer 在大多数数据集上的补全质量优于其余通用时序分析方法, 这可能得益于它们的模型结构, 以及专门针对时序数据补全任务进行设计的损失函数和训练过程. 包括大模型在内的通用时序分析方法在补全任务上的性能与适用性仍有提升空间.

3) 多数方法在 MCAR 缺失机制下有最佳的补全质量, 而在 MAR 尤其是 MNAR 缺失机制下的补全质量明显下降 , 表明现有方法在应对非随机缺失时仍面临挑战.

4) 多数方法在点状缺失模式下有最好的补全质量. 在块状缺失和全变量缺失模式下的补全质量与数据集特性以及补全方法本身的架构有关. 例如, 对于变量间相关性较低的数据集, 块状缺失模式使 RNN 方法的补全难度显著提升. 而在变量相关性较强的数据集中, 全变量缺失模式对基于注意力机制的方法影响更大, RNN 方法反而展现更强的鲁棒性.

5) 在所有数据集中, 缺失率的增加均会导致补全误差上升. 虽然深度学习方法整体优于传统方法, 但在变量维度动态变化的数据集中(如 Electricity), 假设输入维度固定的深度模型可能面临适应性不足的问题, 从而性能下降.

6) 从数据集中挑选出不同数量的变量进行补全会影响补全的质量, 当变量数量增加时, 深度学习方法可以利用变量间关系进行补全, 但随着变量数量继续变大时其边际效益是递减的. 不同数据集上具有最优补全质量的变量数量存在差异, 可能与数据集内变量间的相关性有关.

7) 输入时序的长度过长或过短都会降低大多数深度学习方法的补全质量, 相对于基于注意力的方法, 基于 RNN 的方法更能在输入时序变长时保持补全质量的稳定.

8) 经过深度学习方法进行补全的数据确实能对下游任务产生增益作用, 但是补全效果与增益效果之间存在偏差, 补全误差最小的数据并不一定最有利于下游任务.

6.7 方法推荐

为帮助研究者在实际应用中选择合适的深度学习时序数据补全方法, 本文将各类方法在所有实验数据集上的补全误差和运行时间进行聚合, 分别筛选出在多数数据集上具有较低补全误差的方法(高准确度方法)和具有较短运行时间的方法(高效率方法).

此外, 依据实验结果, 本文归纳出影响补全方法

性能的典型数据特征如下:

1) 高相关性. 数据集中变量间存在高相关性, 如PeMS 和 Solar Alabama 数据集.

2) 高缺失率. 缺失率达到 50% 或超过 50%.

3) block missing. 沿时间维度的连续片段缺失.

4) blackout. 部分时间步上所有变量同时缺失.

5) 非完全随机缺失. 缺失机制为 MAR 或 MNAR.

6) 高维数据. 变量数量或者序列长度较大.

图 21 中的雷达图可直观展现各方法在不同特征的数据上的补全质量. 对于每一种典型数据特征, 计算各方法在该特征数据上的平均补全准确度排名得分: 排名最高得 5 分, 排名最低得 1 分, 越远离中心表明方法在该特征数据上的补全质量越优.

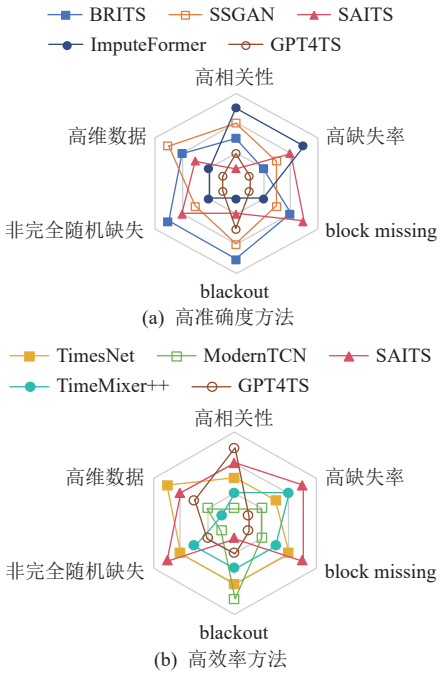


Fig. 21 Imputation quality comparison of deep learning methods across various data features

图 21 深度学习方法在各数据特征上的补全质量比较

7 发展趋势与展望

基于对现有方法的分析及实验结果的启示, 本文对不完整时序数据补全领域的发展趋势提出 6 点展望:

1)面向大规模时序数据的补全方法

在大数据时代, 伴随物联网和传感器技术的广泛应用, 时序数据的规模呈指数级增长. 尽管现有基于深度学习的补全方法在补全质量上表现优异, 但相比统计方法或机器学习方法, 其计算成本较高, 难以直接应用于大规模数据场景. 未来研究可从 2 个方向着手应对这一挑战: 一方面, 降低补全方法的计算复杂度, 例如引入降维技术、稀疏注意力机制[85] 或轻量化网络结构; 另一方面, 提升计算资源利用效率, 如采用分布式训练、多 GPU 并行推理, 或结合联邦学习[90] 实现跨设备补全的同时保护数据隐私. 这些策略将有助于推动深度学习补全方法在实际大规模生产环境中的部署.

2)复杂缺失场景下的时序数据补全方法

现有大多数时序数据补全方法通常基于 MCAR 假设展开, 较少考虑更贴近现实的 MAR 和 MNAR 缺失机制. 然而在实际应用中, 非完全随机的缺失机制广泛存在, 尤其在 MNAR 场景下, 观测数据与完整数据之间可能存在显著的分布偏移[29], 忽视缺失机制易在补全过程中引入误差. 此外, 缺失模式本身往往蕴含有关数据生成机制的信息, 若能有效建模和利用这些信息, 或可显著提升补全性能. 因此, 未来研究应更加关注复杂的缺失场景(如 MAR、MNAR 缺失机制与块状缺失、全变量缺失等模式相结合的情况), 通过对数据中缺失模型的建模, 提升方法在真实数据上的补全质量与鲁棒性.

3)时序数据补全与下游任务的集成优化

时序数据补全的核心目标在于提升数据质量, 从而改善基于数据的下游任务效果. 目前主流方法普遍采用“先补全、后建模”的两阶段流程, 然而补全过程可能引入额外误差, 且补全质量与下游任务表现之间存在错位, 影响整体性能与效率. 将补全与下游任务集成有望缓解上述问题, 如 SSGAN[46] 通过联合补全与分类任务训练模型, 实现了更优的下游表现. 此外, 也有工作尝试直接基于不完整时序数据进行表征学习, 如利用 CNN 提取特征并完成后续任务[91]. 未来, 构建面向不完整时序数据下游任务的端到端建模框架, 有望在提升模型性能的同时降低计算开销与系统复杂度, 为实际应用提供更高效的解决方案.

4)面向时序特性的损失函数与评价指标设计

目前多数时序模型在训练过程中使用 *RMSE*、*MSE* 和 *MAE* 等逐点损失函数, 这类损失函数将每个时间步视为独立的单元, 仅仅关注数值差异, 而忽略了时序数据的整体结构, 这可能使得模型难以准确捕捉时序依赖. 最新的工作尝试构造时序片段级别的损失函数[92] 以提升现有时序预测方法的性能. 另一方面, 常用的评价指标难以反映补全结果对实际应用的影响, 评价指标的创新也能指导研究人员更有效地设计和评估时序数据补全方法. 可以适配不

同模型的损失函数与评价指标的研究工作将是推动时序数据补全方法进步的重要基础性工作.

5)大模型与多模态信息的融合应用

现实世界中的时序数据变化不仅受到数值信息的驱动,还常常伴随着丰富的多模态信息(如气象报告与卫星图像等).然而当前大多数时序数据补全方法仍主要依赖单一模态的数值数据进行建模,忽视了其他模态所蕴含的潜在时序线索,从而限制了补全性能的进一步提升.具备多模态建模能力的大模型有望成为解决该挑战的可能路径.近期已有研究尝试构建文本与时序数据对齐的数据集,用于提升模型在时序预测任务中的性能[93],初步验证了该方向的潜力.然而多模态时序数据集的缺乏仍是该研究方向的一大挑战,未来若能推动构建高质量的多模态对齐(如“文本-时序”“图像-时序”等)的时序数据集,将为多模态时序数据补全研究奠定坚实基础,进而促进模型在复杂真实场景中的泛化与应用.

6)面向在线与轻量化部署场景的补全方法

许多时序任务存在实时性要求(如医疗监测、设备状态感知等场景),对模型的延迟、适应性、稳定性提出了更高要求.在这类流式数据环境中,补全模型需具备实时响应能力,并能适应数据分布的动态变化.未来适用于在线部署的时序数据补全方法可通过滑动窗口机制实现增量推理,避免全序列的重复计算[94],同时结合持续学习策略动态更新模型参数,以提升对分布漂移的鲁棒性.另一方面,轻量化部署的需求也在日益增加,深度学习补全方法在嵌入式设备上运行时面临计算资源和能效的限制,可探索模型剪枝、蒸馏、量化等技术,以及与边缘硬件协同设计的方案,提升其部署可行性.

8 总　　结

不完整时序数据补全是时序分析任务中的关键环节,近年来随着深度学习技术的发展,该领域涌现出大量的优秀方法.本文围绕这一任务展开系统综述,首先对不完整时序数据补全任务中的概念与定义进行简要介绍.随后从建模时序依赖的基础架构出发,将主流的深度学习方法划分为4类:基于RNN、基于CNN、基于注意力机制和基于大模型的方法.针对每一种方法,本文对其技术路线进行总结,分析了其适用的任务类型、输出类型以及在基础架构之上采用的增强技术.在方法综述的基础上,本文进一步整理了该领域常用的开源数据集与评估指标,并对实验设计及其中的缺失场景设计进行了详细说明.实验部分选取了2种统计方法、2种机器学习方法以及10种具有代表性的深度学习方法,在来自多个领域的9个公开数据集上进行综合评估.通过对不同缺失场景、缺失率、变量数量和序列长度下的方法表现进行对比分析,系统揭示了各类方法在补全质量与计算效率方面的差异.同时,本文还将补全结果用于下游分类与预测任务进行验证,进一步评估补全方法对于实际应用效果的影响.在对实验进行总结后,本文归纳了不完整时序数据上的典型特征,并依据这些特征推荐合适的深度学习补全方法.最后,本文展望了时序数据补全领域未来的发展趋势,希望能为后续研究提供有价值的参考.

作者贡献声明:周立负责主要工作,包括文献整理、实验实施与分析及论文撰写;吴洋洋参与研究思路讨论,协助制定论文框架并对撰写提供建议;苗晓晔提出指导意见.

参　考　文　献

[1] Liu Shuai, Qiao Ying, Luo Xiongfei, et al. Key techniques of time series databases: A survey[J]. Journal of Computer Research and Development, 2024, 61(3): 614−638 (in Chinese)
(刘帅, 乔颖, 罗雄飞, 等. 时序数据库关键技术综述[J]. 计算机研究与发展, 2024, 61(3): 614−638)

[2] Song Lingyun, Ma Zhuoyuan, Li Zhanhuai, et al. Review on temporal graph neural networks for financial risk prediction[J]. Journal of Software, 2024, 35(8): 3897−3922 (in Chinese)
(宋凌云, 马卓源, 李战怀, 等. 面向金融风险预测的时序图神经网络综述[J]. 软件学报, 2024, 35(8): 3897−3922)

[3] Li Yaguang, Yu R, Shahabi C, et al. Diffusion convolutional recurrent neural network: Data-driven traffic forecasting[J]. arXiv preprint, arXiv: 1707.01926, 2017

[4] Zhang Shuyi, Guo Bin, Dong Anlan, et al. Cautionary tales on air-quality improvement in Beijing[J]. Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences, 2017, 473(2205): 20170457

[5] Yoon J, Zame W R, Van Der Schaar M. Estimating missing data in temporal data streams using multi-directional recurrent neural networks[J]. IEEE Transactions on Biomedical Engineering, 2018, 66(5): 1477−1490

[6] Kazijevs M, Samad M D. Deep imputation of missing values in time series health data: A review with benchmarking[J]. Journal of Biomedical Informatics, 2023, 144: 104440

[7] Silva I, Moody G, Scott D J, et al. Predicting in-hospital mortality of ICU patients: The physionet/computing in cardiology challenge 2012[C]//Proc of 2012 Computing in Cardiology. Piscataway, NJ:

IEEE, 2012: 245−248

[8] Box G E, Jenkins G M, Reinsel G C, et al. Time Series Analysis: Forecasting and Control[M]. Hoboken, NJ: John Wiley & Sons, 2015

[9] Cover T, Hart P. Nearest neighbor pattern classification[J]. IEEE Transactions on Information Theory, 1967, 13(1): 21−27

[10] Wang Y X, Zhang Y J. Nonnegative matrix factorization: A comprehensive review[J]. IEEE Transactions on Knowledge and Data Engineering, 2012, 25(6): 1336−1353

[11] Quinlan J R. Induction of decision trees[J]. Machine Learning, 1986, 1: 81−106

[12] Elman J L. Finding structure in time[J]. Cognitive Science, 1990, 14(2): 179−211

[13] LeCun Y, Bottou L, Bengio Y, et al. Gradient-based learning applied to document recognition[J]. Proceedings of the IEEE, 2002, 86(11): 2278−2324

[14] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need[C]//Advances in Neural Information Processing Systems 30. San Diego, CA: Neural Information Processing Systems Foundation, Inc., 2017: 5998−6008

[15] Goodfellow I J, Pouget-Abadie J, Mirza M, et al. Generative adversarial nets[C]//Advances in Neural Information Processing Systems 27. San Diego, CA: Neural Information Processing Systems Foundation, Inc., 2014: 2672−2680

[16] Ho J, Jain A, Abbeel P. Denoising diffusion probabilistic models[C]//Advances in Neural Information Processing Systems 33. San Diego, CA: Neural Information Processing Systems Foundation, Inc., 2020: 6840−6851

[17] Scarselli F, Gori M, Tsoi A C, et al. The graph neural network model[J]. IEEE Transactions on Neural Networks, 2008, 20(1): 61−80

[18] Brown T, Mann B, Ryder N, et al. Language models are few-shot learners[C]//Advances in Neural Information Processing Systems 33. San Diego, CA: Neural Information Processing Systems Foundation, Inc., 2020: 1877−1901

[19] Khayati M, Lerner A, Tymchenko Z, et al. Mind the gap: An experimental evaluation of imputation of missing values techniques in time series[J]. Proceedings of the VLDB Endowment, 2020, 13(5): 768−782

[20] Fang Chenguang, Wang Chen. Time series data imputation: A survey on deep learning approaches[J]. arXiv preprint, arXiv: 2011.11347, 2020

[21] Weerakody P B, Wong K W, Wang G, et al. A review of irregular time series data handling with gated recurrent neural networks[J]. Neurocomputing, 2021, 441: 161−178

[22] Cho K, Van Merriënboer B, Gulcehre C, et al. Learning phrase representations using RNN encoder-decoder for statistical machine translation[J]. arXiv preprint, arXiv: 1406.1078, 2014

[23] Brophy E, Wang Zhengwei, She Qi, et al. Generative adversarial networks in time series: A systematic literature review[J]. ACM Computing Surveys, 2023, 55(10): 1−31

[24] Jin Min, Koh H Y, Wen Qingsong, et al. A survey on graph neural networks for time series: Forecasting, classification, imputation, and anomaly detection[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2024, 46(12): 10466−10485

[25] Jin Min, Wen Qingsong, Liang Yuxuan, et al. Large models for time series and spatio-temporal data: A survey and outlook[J]. arXiv preprint arXiv: 2310.10196, 2023

[26] Lin Lequan, Li Zhengkun, Li Ruikun, et al. Diffusion models for time-series applications: A survey[J]. Frontiers of Information Technology & Electronic Engineering, 2024, 25(1): 19−41

[27] Wang Jun, Du Wenjie, Yang Yiyuan, et al. Deep learning for multivariate time series imputation: A survey[J]. arXiv preprint arXiv: 2402.04059, 2024

[28] Pesaran M H. Time Series and Panel Data Econometrics[M]. Oxford, UK: Oxford University Press, 2015

[29] Van Buuren S. Flexible Imputation of Missing Data[M]. Boca Raton, FL: CRC Press, 2018

[30] Rubin D B. Inference and missing data[J]. Biometrika, 1976, 63(3): 581−592

[31] Shao Jun, Zhong B. Last observation carry - forward and last observation analysis[J]. Statistics in Medicine, 2003, 22(15): 2429−2441

[32] Zhang Aoqiang, Song Shaoxu, Wang Jianmin, et al. Time series data cleaning: From anomaly detection to anomaly repairing[J]. Proceedings of the VLDB Endowment, 2017, 10(10): 1046−1057

[33] Van Buuren S, Groothuis-Oudshoorn K. MICE: Multivariate imputation by chained equations in R[J]. Journal of Statistical Software, 2011, 45: 1−67

[34] Zhang Aoqiang, Song Shuxu, Sun Yu, et al. Learning individual models for imputation[C]//Proc of 2019 IEEE 35th Int Conf on Data Engineering (ICDE). Piscataway, NJ: IEEE, 2019: 160−171

[35] Wellenzohn K, Böhlen M, Dignös A, et al. Continuous imputation of missing values in streams of pattern-determining time series[C]// Proc of Int Conf on Extending Database Technology. Berlin: Springer, 2017: 330−341

[36] Yi Xiuwen, Zheng Yu, Zhang Junbo, et al. ST-MVL: Filling missing values in geo-sensory time series data[C]//Proc of the 25th Int Joint Conf on Artificial Intelligence. Palo Alto, CA: AAAI, 2016: 2704−2710

[37] Rahman M G, Islam M Z. A decision tree-based missing value imputation technique for data pre-processing[C]//Proc of the 9th Australasian Data Mining Conf (AusDM 2011). Sydney, New South Wales: Australia Australian Computer Society Inc, 2011: 41−50

[38] Zhang Xinmeng, Yan Chao, Gao Cheng, et al. Predicting missing values in medical data via XGBoost regression[J]. Journal of Healthcare Informatics Research, 2020, 4: 383−394

[39] Mazumder R, Hastie T, Tibshirani R. Spectral regularization algorithms for learning large incomplete matrices[J]. The Journal of Machine Learning Research, 2010, 11: 2287−2322

[40] Yu H F, Rao N, Dhillon I S. Temporal regularized matrix factorization for high-dimensional time series prediction[C]//Advances in Neural Information Processing Systems 29. San Diego, CA: Neural Information Processing Systems Foundation, Inc., 2016: 847−855

[41] Che Zhengping, Purushotham S, Cho K, et al. Recurrent neural networks for multivariate time series with missing values[J]. Scientific Reports, 2018, 8(1): 6085

[42] Cao Wei, Wang Dong, Li Jian, et al. BRITS: Bidirectional recurrent imputation for time series[C]//Advances in Neural Information Processing Systems 31. San Diego, CA: Neural Information Processing Systems Foundation, Inc., 2018: 6776−6786

[43] Luo Yonghong, Cai Xiangrui, Zhang Ying, et al. Multivariate time series imputation with generative adversarial networks[C]//Advances in Neural Information Processing Systems 31. San Diego, CA: Neural Information Processing Systems Foundation, Inc., 2018: 1603−1614

[44] Luo Yonghong, Zhang Ying, Cai Xiangrui, et al. E²GAN: End-to-end generative adversarial network for multivariate time series imputation [C]// Proc of the 28th Int Joint Conf on Artificial Intelligence. Palo Alto, CA: AAAI, 2019: 3094−3100

[45] Liu Yukai, Yu R, Zheng S, et al. NAOMI: Non-autoregressive multiresolution sequence imputation[C]//Advances in Neural Information Processing Systems 32. San Diego, CA: Neural Information Processing Systems Foundation, Inc., 2019: 11236−11246

[46] Miao Xiaoye, Wu Yangyang, Wang Jun, et al. Generative semi-supervised learning for multivariate time series imputation[C]//Proc of the AAAI Conf on Artificial Intelligence. Palo Alto, CA: AAAI, 2021: 8983−8991

[47] Fortuin V, Baranchuk D, Rätsch G, et al. GP-VAE: Deep probabilistic time series imputation[C/OL]//Proc of Int Conf on Artificial Intelligence and Statistics. PMLR, 2020: 1651−1661. [2025-06-05]. https://proceedings.mlr.press/v108/fortuin20a.html

[48] Wu Haixu, Hu Tengge, Liu Yong, et al. TimesNet: Temporal 2D-variation modeling for general time series analysis[C/OL]//Proc of the 11th Int Conf on Learning Representations. 2023[2025-06-05]. https://openreview.net/forum?id=ju_Uqw384Oq

[49] Luo Donghao, Wang Xue. ModernTCN: A modern pure convolution structure for general time series analysis[C/OL]// Proc of the 12th Int Conf on Learning Representations. 2024[2025-06-05]. https://openreview.net/forum?id=vpJMJerXHU

[50] Bansal P, Deshpande P, Sarawagi S. Missing value imputation on multidimensional time series[J]. Proceedings of the VLDB Endowment, 2021, 14(11): 2533−2545

[51] Tashiro Y, Song Jiaming, Song Yang, et al. CSDI: Conditional score-based diffusion models for probabilistic time series imputation[C]// Advances in Neural Information Processing Systems 34. San Diego, CA: Neural Information Processing Systems Foundation, Inc., 2021: 24804−24816

[52] Marisca I, Cini A, Alippi C. Learning to reconstruct missing data from spatiotemporal graphs with sparse observations[C]//Advances in Neural Information Processing Systems 35. San Diego, CA: Neural Information Processing Systems Foundation, Inc., 2022: 32069−32082

[53] Liu Mingzhe, Huang Han, Feng Hao, et al. PriSTI: A conditional diffusion framework for spatiotemporal imputation[C]// Proc of 2023 IEEE 39th Int Conf on Data Engineering (ICDE). Piscataway, NJ: IEEE, 2023: 1927−1939

[54] Du Wenjie, Côté D, Liu Yan. SAITS: Self-attention-based imputation for time series[J]. Expert Systems with Applications, 2023, 219: 119619

[55] Wu Qiang, Yao Gechang, Feng Zhixi, et al. Peri-midFormer: Periodic pyramid Transformer for time series analysis[C]// Advances in Neural Information Processing Systems 37. San Diego, CA: Neural Information Processing Systems Foundation, Inc., 2024: 13035−13073

[56] Nie Tong, Qin Guoyang, Ma Wei, et al. ImputeFormer: Low rankness-induced transformers for generalizable spatiotemporal imputation [C]//Proc of the 30th ACM SIGKDD Conf on Knowledge Discovery and Data Mining. New York: ACM, 2024: 2260−2271

[57] Yang Xinyu, Sun Yu, Chen Xiaojie. Frequency-aware generative models for multivariate time series imputation[C]//Advances in Neural Information Processing Systems 37. San Diego, CA: Neural Information Processing Systems Foundation, Inc., 2024: 52595−52623

[58] Yuan Xinyu, Qiao Yan. Diffusion-TS: Interpretable diffusion for general time series generation[C/OL]// Proc of the 12th Int Conf on Learning Representations. [2025-06-05]. https://openreview.net/forum?id=4h1apFjO99

[59] Zhang Yunhao, Liu Minghao, Zhou Shengyang, et al. UP2ME: Univariate pre-training to multivariate fine-tuning as a general-purpose framework for multivariate time series analysis[C/OL]// Proc of Int Conf on Machine Learning. PMLR, 2024: 59358−59381. [2025-06-05]. https://openreview.net/forum?id=aR3uxWlZhX

[60] Wang Shiyu, Li Jiawei, Shi Xiaoming, et al. TimeMixer++: A general time series pattern machine for universal predictive analysis[J]. arXiv preprint, arXiv: 2410.16032, 2024

[61] Zhou Tian, Niu Peisong, Wang Xue, et al. One fits all: Power general time series analysis by pretrained LM[C]// Advances in Neural Information Processing Systems 36. San Diego, CA: Neural Information Processing Systems Foundation, Inc., 2023: 43322−43355

[62] Chen C, Oliveira G L, Sharifi-Noghabi H, et al. Enhance time series modeling by integrating LLM[C/OL]//NeurIPS Workshop on Time Series in the Age of Large Models. San Diego, CA: Neural Information Processing Systems Foundation Inc. 2024[2025-06-05] https://openreview.net/pdf?id=LN6KzMLKOJ

[63] Goswami M, Szafer K, Choudhry A, et al. MOMENT: A family of open time-series foundation models[C/OL]// Proc of Int Conf on Machine Learning. PMLR, 2024: 16115−16152. [2025-06-05]. https://openreview.net/forum?id=FVvf69a5rx

[64] Liu Yong, Zhang Haoran, Li Chenyu, et al. Timer: Generative pre-trained transformers are large time series models[C/OL]// Proc of Int Conf on Machine Learning. PMLR, 2024: 32369−32399. [2025-06-05]. https://openreview.net/forum?id=bYRYb7DMNo

[65] Talukder S J, Yue Y, Gkioxari G. TOTEM: Tokenized time series embeddings for general time series analysis[J/OL]. Transactions on Machine Learning Research, 2024[2025-06-05]. https://openreview.net/forum?id=QlTLkH6xRC

[66] Chen Tianqi, Guestrin C. XGBoost: A scalable tree boosting system[C]//Proc of the 22nd ACM SIGKDD Int Conf on Knowledge Discovery and Data Mining. New York: ACM, 2016: 785−794

[67] Bengio Y, Courville A, Vincent P. Representation learning: A review and new perspectives[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2013, 35(8): 1798−1828

[68] Sutskever I. Sequence to sequence learning with neural networks[J]. arXiv preprint, arXiv: 1409.3215, 2014

[69] Bai Shaojie, Kolter J Z, Koltun V. An empirical evaluation of generic

convolutional and recurrent networks for sequence modeling[J]. arXiv preprint, arXiv: 1803.01271, 2018

[70] Long J, Shelhamer E, Darrell T. Fully convolutional networks for semantic segmentation[C]//Proc of the IEEE Conf on Computer Vision and Pattern Recognition. Los Alamitos, CA: IEEE Computer Society, 2015: 3431−3440

[71] Wei J, Tay Y, Bommasani R, et al. Emergent abilities of large language models[J/OL]. Transactions on Machine Learning Research, 2022[2025-06-05]. https://openreview.net/forum?id=yzkSU5zdwD

[72] Touvron H, Lavril T, Izacard G, et al. LLaMA: Open and efficient foundation language models[J]. arXiv preprint, arXiv: 2302.13971, 2023

[73] Achiam J, Adler S, Agarwal S, et al. Gpt-4 technical report[J]. arXiv preprint, arXiv: 2303.08774, 2023

[74] Rumellhart D E. Learning internal representations by error propagation[J]. Parallel Distributed Processing: Explorations in the Microstructure of Cognition, 1986, 1: 319−362

[75] Cini A, Marisca I, Alippi C. Filling the G_ap_s: Multivariate time series imputation by graph neural networks[C/OL]// Proc of Int Conf on Learning Representations, 2022[2025-06-05]. https://openreview.net/forum?id=kOu3-S3wJ7

[76] Rumelhart D E, Hinton G E, Williams R J. Learning representations by back-propagating errors[J]. Nature, 1986, 323(6088): 533−536

[77] Kingma D P, Welling M. Auto-encoding variational Bayes[J]. arXiv preprint, arXiv: 1312.6114, 2013

[78] Williams C K, Rasmussen C E. Gaussian Processes for Machine Learning: Vol 2[M]. Cambridge, MA: MIT Press, 2006

[79] Szegedy C, Liu Wei, Jia Yangqing, et al. Going deeper with convolutions[C]//Proc of the IEEE Conf on Computer Vision and Pattern Recognition. Los Alamitos, CA: IEEE Computer Society, 2015: 1−9

[80] Radford A, Wu J, Child R, et al. Language models are unsupervised multitask learners[J]. OpenAI Blog, 2019, 1(8): 9

[81] Devlin J, Chang M W, Lee K, et al. BERT: Pre-training of deep bidirectional transformers for language understanding[C]//Proc of the 2019 Conf of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Stroudsburg, PA: ACL, 2019: 4171−4186

[82] Bao Hangbo, Dong Li, Piao Songhao, et al. BEiT: BERT pre-training of image transformers[C/OL]// Proc of Int Conf on Learning Representations, 2022[2025-06-05]. https://openreview.net/forum?id=p-BhZSz59o4

[83] Kraskov A, Stögbauer H, Grassberger P. Estimating mutual information[J]. Physical Review E—Statistical, Nonlinear, and Soft Matter Physics, 2004, 69(6): 066138

[84] De Vito S, Massera E, Piga M, et al. On field calibration of an electronic nose for benzene estimation in an urban pollution monitoring scenario[J]. Sensors and Actuators B: Chemical, 2008, 129(2): 750−757

[85] Zhou Haoyi, Zhang Shanghang, Peng Jieqi, et al. Informer: Beyond efficient transformer for long sequence time-series forecasting [C]//Proc of the AAAI Conf on Artificial Intelligence. Palo Alto, CA: AAAI, 2021, 35(12): 11106−11115

[86] Reyna M A, Josef C S, Jeter R, et al. Early prediction of sepsis from clinical data: The physionet/computing in cardiology challenge 2019[J]. Critical Care Medicine, 2020, 48(2): 210−217

[87] Du Wenjie. Pypots: A Python toolbox for data mining on partially-observed time series[J]. arXiv preprint, arXiv: 2305.18811, 2023

[88] Kingma D P, Ba J. Adam: A method for stochastic optimization[J]. arXiv preprint, arXiv: 1412.6980, 2014

[89] Hosmer Jr D W, Lemeshow S, Sturdivant R X. Applied Logistic Regression[M]. Hoboken, NJ: John Wiley & Sons, 2013

[90] Chen Shengchao, Long Guodong, Jiang Jing, et al. Federated foundation models on heterogeneous time series[C]//Proc of the AAAI Conf on Artificial Intelligence. Palo Alto, CA: AAAI, 2025, 39(15): 15839−15847

[91] Zhang Kai, Li Chao, Yang Qinmin. TriD-MAE: A generic pre-trained model for multivariate time series with missing values[C]//Proc of the 32nd ACM Int Conf on Information and Knowledge Management. New York: ACM, 2023: 3164−3173

[92] Kudrat D, Xie Zongxia, Sun Yanru, et al. Patch-wise structural loss for time series forecasting[J]. arXiv preprint, arXiv: 2503.00877, 2025

[93] Liu Haoxin, Xu Shangqing, Zhao Zhiyuan, et al. Time-MMD: Multi-domain multimodal dataset for time series analysis[C]// Advances in Neural Information Processing Systems 37. San Diego, CA: Neural Information Processing Systems Foundation, Inc., 2024: 77888−77933

[94] Chen Xie, Wu Yu, Wang Zhenghao, et al. Developing real-time streaming transformer transducer for speech recognition on large-scale dataset[C]//Proc of IEEE Int Conf on Acoustics, Speech and Signal Processing (ICASSP2021). Piscataway, NJ: IEEE, 2021: 5904−5908



Zhou Li, born in 1993. Master candidate. His main research interest includes time series imputation and forecasting.

周　立, 1993 年生. 硕士研究生. 主要研究方向为时序数据补全与预测.



Wu Yangyang, born in 1994. PhD. Distinguished research fellow. His main research interests include large language models, data mining, data imputation, and data cleaning.

吴洋洋, 1994 年生. 博士. 特聘研究员. 主要研究方向为大语言模型、数据挖掘、数据补全、数据清洗.



Miao Xiaoye, born in 1989. PhD, associate professor. Senior member of CCF. Her main research interest includes intelligent data governance and services.

苗晓晔, 1989 年生. 博士, 副教授. CCF 高级会员. 主要研究方向为智能数据治理与服务.