文章编号：2095-2163(2022)04-0135-05 中图分类号：TP319 文献标志码：A

降低数据稀疏性的多维时序序列时间戳对齐方法

李广盛[1]，郑建立[1]，车霞静[2]
（1 上海理工大学 健康科学与工程学院，上海 200093；2 上海交通大学附属仁济医院，上海 200127）

摘 要： 多维时序序列是指一组按照时间发生先后顺序进行排列的数据点序列，广泛存在于天文、医疗、交通等领域。囿于收集技术较差，或是序列的物理性质所致，时序序列记录中往往存在较多的缺失值和大量的不规则采样，使得时序序列的稀疏性大大增加。最终导致许多深度学习的时序序列分类算法等无法正常工作，出现算法效果差、算法训练时间过长等问题。面对这些问题，目前常用的做法是简单删减或是利用专家知识做重采样，前者会导致数据规模变小，后者使得算法成本增加。本文利用时序序列的时间戳数据构建了一种半自动化的预处理方法。在公共数据集 MIMIC-III、Physionet 和肾移植数据集上的实验表明本文提出的方法在基本不损失算法效果的同时，能够有效降低数据稀疏规模，并且平均能够节约 42.1%的算法训练时间。

关键词： 多维时序序列分类；深度学习；缺失值；不规则采样

Timestamp alignment method of multi-dimensional time series sequence to reduce data sparsity

LI Guangsheng[1], ZHENG Jianli[1], CHE Xiajing[2]
(1 School of Health Science and Engineering, University of Shanghai for Science and Technology , Shanghai 200093, China; 2 Renji Hospital Affiliated to Shanghai Jiao Tong University, Shanghai 200127, China)

【Abstract】 Multi-dimensional time series refers to a series of data points arranged in the order of time, which is widely used in astronomy, medical treatment, transportation and other fields. Due to poor collection technology or the physical properties of the sequence, there are often more missing values and a large number of irregular sampling in the time series sequence record, which greatly increases the sparsity of the time series sequence. In the end, many deep learning time series sequence classification algorithms cannot work normally, and problems such as poor algorithm effect and long algorithm-training time occur. In the face of these problems, the current common method is to simply delete or use expert knowledge to do resampling. The former will result in a smaller data size, and the latter will increase the cost of the algorithm. In this paper, a semi-automatic preprocessing method is constructed using the timestamp data of the time series sequence. Experiments on the public data set MIMIC-III, Physionet and kidney transplantation data set show that the method proposed in this paper can effectively reduce the sparse scale of the data while basically not losing the effect of the algorithm, and can save the algorithm training time on average by 42.1%.

【Key words】 multivariate time series classification; deep learning; missing values; irregular sampling

0 引 言

在过去的二十年中，时间序列分类(time series classification，TSC)被认为是数据挖掘中最具挑战性的问题之一[1-2]。随着时间数据可用性的增加，自 2015 年以来已有数百种 TSC 算法被提出[3]。由于时间序列数据的自然时序性，几乎每一个需要某种人类认知过程的任务中都会出现时间序列数据[4]。时间序列广泛存在各类研究工作中，包括电子健康记录[5]、人类活动识别[6]到声学场景分类[7]和网络安全[8]等领域。但由于种种原因，如收集错误、故意损坏、医疗事件、节省成本、设备异常等，往往会不可避免地出现丢失观测数据和不规则采样等现象，使得时序序列数据稀疏性大大增加，阻碍了分类任务的开展。

针对时序序列中缺失问题，从不同的解决方法来看，主要可以分为 2 类。一是以专家知识为基础进行手工填补和重采样；二是利用深度学习等方法实现端到端的数据填补及分类。前者主要利用专家知识，根据时序序列数据的观测变量等信息进行缺失值的填补和修正[9-10]，后者利用深度学习强大的抽象表征能力和拟合能力来实现数据的填补和分类[11-14]。

基于专家知识的方法尽管可解释性较强，但是

作者简介：李广盛(1995-)，男，硕士研究生，主要研究方向：医学信息系统集成技术、医学人工智能；郑建立(1965-)，男，博士，副教授，硕士生导师，主要研究方向：医学信息系统与集成技术、医学人工智能、医学仪器嵌入式控制系统。
通讯作者：郑建立 Email：zhengjianli163@ 163.com

收稿日期：2021-11-19

却费时费力;而基于深度学习方法在原始数据集上直接填补尽管能够取得不错的效果,但是却忽视了不规则采样等问题。此外,数据集中可能存在部分数据缺失率过高,使得模型无法抽取其潜在信息,模型的填补效果大打折扣。本文提出一种基于数据集中自带的时间戳数据,通过数据时间戳对齐和下采样方法,在多个公开数据集以及私有数据集和近年来提出的深度学习时序序列分类算法上的实验表明,该方法能够在基本不损失模型效果的同时,有效减小数据集的稀疏规模和模型训练时间。

1 相关方法

在本节中,本文先给出多维时序序列的相关定义,之后将相关方法分为时间戳对齐和基于分布密度的下采样两步讲述,具体流程示意图如图 1 所示。

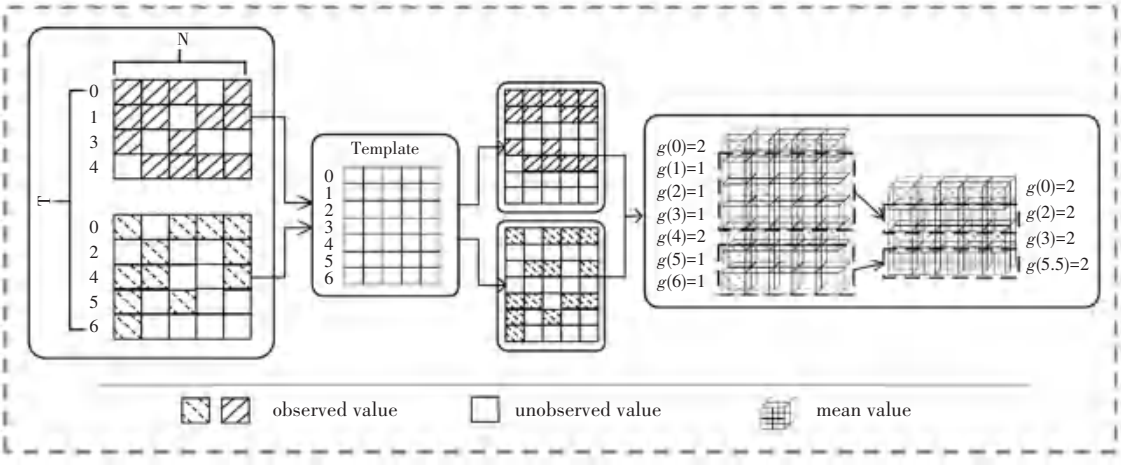

图 1 时间戳对齐和下采样流程示意图

Fig. 1 Schematic diagram of time stamp alignment and downsampling process

1.1 多维时序序列的定义

本文将一个具有 N 个观测变量和 T 个观测时间点的时序序列定义为 $\boldsymbol{X}=(\boldsymbol{x}_{t_0},\boldsymbol{x}_{t_1},\cdots,\boldsymbol{x}_{t_{T-1}})\in R^{T\times N}$,其时间戳定义为 $\boldsymbol{t}=(t_0,t_1,\cdots,t_{T-1})\in R^{\mathrm{T}}$。因此,$x_{t_i}^j$ 代表 $\boldsymbol{x}_{t_i}$ 的第 j 个观测变量,$\boldsymbol{x}_{t_i}$ 称为一次观测。本文另外定义一个掩膜(mask)矩阵 $\boldsymbol{M}\in R^{T\times N}$ 用来表示 $x_{t_i}^j$ 是否为缺失值,其计算公式如下:

$$M_{t_i}^j=\begin{cases}1 & \text{if } x_{t_i}^j \text{is observed}\\ 0 & \text{if } x_{t_i}^j \text{is not observed}\end{cases} \tag{1}$$

1.2 时间戳对齐

由于数据集的不规则采样,导致虽然数据采样点的时间跨度非常大,但是数据点的个数却非常少,具体到每一个样本更是不尽相同。例如在 Physionet 数据集中,总共有 48×60 min,共 2 880 个数据可采样点。但事实上该数据集中最大样本的数据采样点个数只有 249,而最小样本的数据采样点个数只有 1。考虑到深度学习模型在训练时一般采用小批量(mini-batch)做法,因此需要在较短的样本尾部填充无意义的屏蔽值(mask value),使模型的输入等长。但是这样的对齐在 RNN 模型中是有缺陷的,RNN 模型的每一个时刻输入是 mini-batch 在时间维上的切片,上述做法会使得切片中包含的不同样本数据点没有对齐,即样本 A 的 t_i 时刻的数据和样本 B 的 t_j 时刻数据同时输入 RNN 模型,这样会导致模型效果欠佳。因此,需要做数据对齐。

首先本文根据时间戳的最小粒度和其时间跨度,构建一个具有最长数据点长度的无值背景板,再根据原始数据对应的时间戳将每一个数据点嵌入其中,这样就得到了一个完整的所有样本数据点都对齐了的数据集,实现了数据点的物理位置和逻辑位置的统一。根据上述做法,Physionet 数据集的维度从原始的 3 994×203×41,最终则转换成了 3 994×2 881×41。

1.3 基于数据分布密度的下采样

在将数据对齐后,数据集的稀疏性会进一步扩大,需要做进一步的处理来减小数据集的稀疏性。本文定义在时间轴上的数据集分布密度函数,具体如下:

$$g(t_i)=\sum_{k=1}^{K}\sigma(x_{t_i}) \tag{2}$$

其中,K 为数据集样本个数。研究中还推得 $\sigma(x_{t_i})$ 的数学定义公式可写为:

$$\sigma(x_{t_i}) = \begin{cases} 1 & \text{if } \sum_{j=1}^{N} M_{t_i}^{j} > 0 \\ 0 & \text{if } \sum_{j=1}^{N} M_{t_i}^{j} = 0 \end{cases} \tag{3}$$

根据定义可知，当 $g(t)$ 较小时，说明样本在对应时间戳 $t \in [t_i, t_j)$ 中分布较少，该区间的稀疏性较大。本文通过求解该区间所有观测变量的均值来替代该稀疏区域，实现数据稀疏性的减小，计算公式如下：

$$x_{t_c} = \sum_{p=i}^{j} \frac{x_{t_p}}{j - i + 1} \tag{4}$$

其中，t_c 可用如下数学公式计算得出：

$$t_c = \sum_{p=i}^{j} \frac{t_p}{j - i + 1} \tag{5}$$

图 2 给出了 Physionet 数据集原始和预处理后的数据密度分布图像。从图 2 中可以明显看出，经过预处理的数据在时间轴上的分布密度显著提升，并且基本保留原始分布密度的分布趋势。

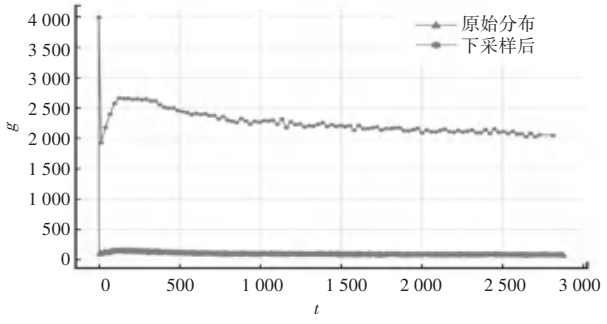

图 2　Physionet 数据集数据分布密度

Fig. 2　Data distribution density of Physionet data set

在经过预处理后，Physionet 数据集大小从经过时间戳对齐后的 3 994×2 881×41 转换成了 3 994×100×41。对比该数据集原始的大小可以发现，经过处理后的 Physionet 数据集的大小是原来的 0.493 倍，显著减少了数据集的尺寸。

2　实验结果

2.1　数据集

Physionet challenge 2012[15] 是 physionet.org 在 2012 年举办的一个多维时序序列分类和回归比赛。该比赛使用的数据是 12 000 名因心脏病、内科、外科等原因而住院的 ICU 病人的记录，包括白蛋白(Albumin)、碱性磷酸酶(ALP)、谷丙转氨酶(ALT)等 36 个观测变量和年龄、身高、体重等 6 个一般描述符，共 42 个变量。除一般描述符外，囿于病人身体状态差以及医疗设备工作性质等原因，在 36 个观测变量中有很多缺失值，且每一个观测的时间间隔也不相同。数据集中给出了每一个观测的相关时间戳，该时间戳的分度值是分钟，即时间的最小粒度为每分钟。该挑战赛设立了 5 个分类任务和一个回归任务。本文主要使用的是其中的死亡预测任务，即预测病人在 48 h 后是否死亡。这也是下文涉及的算法在提出时被使用到的任务。

MIMIC-III Clinical DataBase[16-17] 是一个大型的公开数据库，其中包括了 2001 年至 2012 年期间在美国 BIDMC 医疗中心重症监护病房住院的超过 4 万名患者的已确认的健康相关数据。该数据库包括人口统计信息、在床边进行的生命体征观测、实验室检测结果、程序、药物、护理记录、影像报告和死亡率等记录。通过数据挖掘、信息提取等手段，从该数据库中提取了 X 份存在大量缺失值和不规则采样的 ICU 住院病人 48 h 内的时序序列数据、对应的时间戳和死亡预测标签。该数据一共有 12 个观测变量，包括血氧饱和度(SpO2)、心率(HR)、呼吸速率(RR)、收缩压(SBP)等。和 Physionet 一样，本文也是使用其作为分类任务。

肾移植术后数据集是来自某三甲医院肾移植科的 931 名肾移植患者术后生理检查的数据集，其中包括血常规、尿常规和血药浓度等共 87 个观测变量。该数据集的时间戳较为特殊，以肾移植手术当天为第零天，手术后所做检查的时间戳都为正整数，手术前所做检查的时间戳皆为负整数，时间戳的单位长度为一天。一般肾移植患者术后需住院几周，因此，数据在第零天周围分布比较密集。之后因病人经济原因、个人意愿以及地域等因素，使得病人做生理检查次数较少、检查范围不全，从而导致数据分布十分稀疏且不规则。该数据集的标签分为感染、排异和正常三个类型，分别描述了病人肾移植术后自身免疫力水平低、高、正常对移植肾的影响。

图 3 给出了上述 3 个数据集原始缺失率和经过下采样后的缺失率。从图 3 中可以发现，肾移植数据集缺失率较另外 2 个数据集缺失率更高，下采样效果不明显，但是对于 Physionet 数据集和 MIMIC-III数据集，下采样均有效降低了数据集的缺失率。

2.2　相关分类算法

GRUD[12]，全称 GRU-deacy。文献[12]通过分析缺失值的类型给出了 2 个缺失模式，分别是：固定缺失值模式和衰减收敛缺失值模式。其中，固定缺失值模式指某个观测变量的缺失值和该观测变量最

早的记录值相同;衰减收敛缺失值模式指观测变量在经过较长时间变化后逐渐收敛,如 MIMIC-III 中 SpO2 等观测变量。研究中根据这 2 种缺失值模式提出了填补函数,并将填补过程嵌入普通 GRU 模型,构建了一个端到端的对具有缺失值和不规则采样的多维时序序列进行分类的深度学习算法,在原始 Physionet 数据集实验表明,该算法能够有效地实现对病人死亡与否的预测,其 *AUC* 达到了 0.831,是一个强有力的基线。

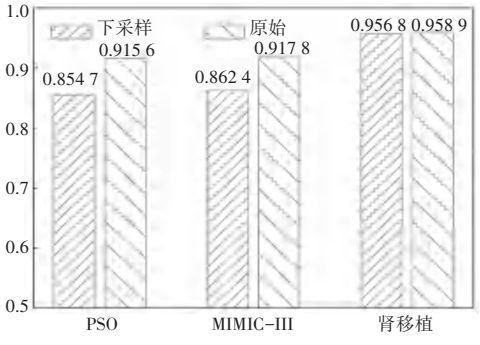

图 3 3 个数据集下采样前后缺失率对比图

Fig. 3 Comparison of missing rates among three data sets with and without downsampling

Interp-net[14] 通过构建了一个插值网络来捕获输入数据的平滑趋势、瞬态和观测强度信息共三个维度的信息,以适应使用稀疏和不规则采样数据作为有监督学习输入的复杂性,从而得到一个规则间隔和无缺失值的输出,在此基础上将利用预测网络计算出最后的分类结果。与 GRUD 不同的是,该模型完全是模块化的,其插值网络和预测网络是分开的。在原始 MIMIC-III 数据集上 *AUC* 达到了0.853。

2.3 结果

由于 3 个数据集标签分布并不均匀,因此本文采用 ROC 曲线下面积 *AUC* 来衡量模型的效果。*AUC* 的计算方法同时考虑了分类器对于正例和负例的分类能力,在样本不平衡的情况下,依然能够对分类器做出合理的评价。实验中将数据集分为训练集、验证集、测试集,其比例为 0.64:0.16:0.2。模型超参数均为模型研发者提供的默认值,其中,Physionet 数据集和肾移植数据集的批次大小为 128,MIMIC-III 批次大小为 256。

表 1 显示了上述模型在 3 个原始数据集和预处理后训练的最终效果。从表 1 中可以看出,模型在经过预处理的数据集上的效果几乎同模型在原始数据集上效果相同,*AUC* 损耗在 0.003。

表 1 GRUD、Interp-net 模型在 Physionet、MIMIC-III、肾移植数据集上 *AUC* 效果表

Tab. 1 *AUC* effect table of GRUD and Interp-net models on Physionet, MIMIC-III, and kidney transplantation data sets

	Physionet		MIMIC-III		肾移植	
	原始	处理后	原始	处理后	原始	处理后
GRUD	0.831	0.823	0.835	0.829	0.592	0.589
Interp-net	0.849	0.843	0.853	0.848	0.625	0.611

本文还对比了上述模型在这 2 类数据集上训练所需时间,所有训练内容都在一张 Nvidia Tesla P40 显卡上进行。实验结果见表 2,单位为 hour/epoch。从表 2 中可以明显看出模型在经过预处理的数据集上达到收敛点的时间较短,能够有效地缩短模型的训练时间:在相同模型情况下,经过处理后的数据集的训练时间与原始数据集训练时间相比,平均减少了 42.1%。尤需指出的是,肾移植数据集在 GRUD 算法上则减少了 50%。

表 2 GRUD、Interp-net 模型在 Physionet、MIMIC-III、肾移植数据集上训练时间表

Tab. 2 Training schedule of GRUD and Interp-net models on Physionet, MIMIC-III, and kidney transplantation datasets

hour · epoch^{-1}

	Physionet		MIMIC-III		肾移植	
	原始	处理后	原始	处理后	原始	处理后
GRUD	0.09	0.05	0.11	0.07	0.08	0.04
Interp-net	0.17	0.11	0.18	0.10	0.12	0.07

3 结束语

本文提出了一种新的多维时序序列预处理方法。首先利用数据集自带的时间戳,实现原始数据在时间刻度上的对齐;然后通过观察数据集在时间轴上的分布密度来缩小分布密度较低的区间,最终

得到一个规则采样且数据稀疏性大大减少的新数据集。实验结果显示与原始数据集相比，在基本不损失模型效果的情况下，该方法显著减小了模型训练所需要的时间。但是，该方法不够自动化，仍需要手动选择需要缩小的区间。因此，性能上更为优越的自动化是未来探索的方向。

参考文献

[1] YANG Qiang, WU Xindong. 10 challenging problems in data mining research [J]. International Journal of Information Technology & Decision Making, 2006, 5(4): 597-604.

[2] ESLING P, AGON C. Time-series data mining [J]. ACM Computing Surveys (CSUR), 2012, 45(1): 12-46.

[3] BAGNALL A, LINES J, BOSTROM A, et al. The great time series classification bake off: A review and experimental evaluation of recent algorithmic advances [J]. Data Mining and Knowledge Discovery, 2017, 31(3): 606-660.

[4] LÄNGKVIST M, KARLSSON L, LOUTFI A. A review of unsupervised feature learning and deep learning for time-series modeling [J]. Pattern Recognition Letters, 2014, 42: 11-24.

[5] RAJKOMAR A, OREN E, CHEN K, et al. Scalable and accurate deep learning with electronic health records [J]. NPJ Digital Medicine, 2018, 1(1): 1-10.

[6] NWEKE H F, TEH Y W, AL-GARADI M A, et al. Deep learning algorithms for human activity recognition using mobile and wearable sensor networks: State of the art and research challenges [J]. Expert Systems with Applications, 2018, 105: 233-261.

[7] NWE T L, DAT T H, MA B. Convolutional neural network with multi-task learning scheme for acoustic scene classification [C]// 2017 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC). Honolulu, HI, USA: IEEE, 2017: 1347-1350.

[8] SUSTO G A, CENEDESE A, TERZI M. Time-series classification methods: Review and applications to power systems data [M]//ARGHANDEH R, ZHOU Yuxun. Big data application in power systems. USA: Elsevier Inc., 2018: 179-220.

[9] KARIM F, MAJUMDAR S, DARABI H, et al. LSTM fully convolutional networks for time series classification [J]. IEEE access, 2017, 6: 1662-1669.

[10] LIPTON Z C, KALE D C, ELKAN C, et al. Learning to diagnose with LSTM recurrent neural networks [J]. arXiv preprint arXiv: 1511.03677, 2015.

[11] LI S C X, MARLIN B. Learning from irregularly-sampled time series: A missing data perspective [C]//International Conference on Machine Learning. PMLR, 2020: 5937-5946.

[12] CHE Z, PURUSHOTHAM S, CHO K, et al. Recurrent neural networks for multivariate time series with missing values [J]. Scientific reports, 2018, 8(1): 1-12.

[13] LUO Yonghong, CAI Xiangrui, ZHANG Ying, et al. Multivariate time series imputation with generative adversarial networks [C]//Proceedings of the 32nd International Conference on Neural Information Processing Systems. Montréal Canada: NIPS, 2018: 1603-1614.

[14] SHUKLA S N, MARLIN B M. Interpolation-prediction networks for irregularly sampled time series [J]. arXiv preprint arXiv: 1909.07782, 2019.

[15] SILVA I, MOODY G, SCOTT D J, et al. Predicting in-hospital mortality of ICU patients: The Physionet/computing in cardiology challenge 2012 [C]//2012 Computing in Cardiology. Krakow, Poland: IEEE, 2012: 245-248.

[16] CHARLES D, GABRIEL M, FURUKAWA M F. Adoption of electronic health record systems among US non-federal acute care hospitals: 2008-2014 [J]. ONC data brief, 2013, 9: 1-9.

[17] COLLINS F S, TABAK L A. Policy: NIH plans to enhance reproducibility [J]. Nature News, 2014, 505(7485): 612-613.

(上接第 134 页)

[6] CHARLES R Q, SU Hao, MO Kaichun, et al. PointNet: Deep learning on point sets for 3D classifification and segmentation [C]// 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Honolulu, HI, USA: IEEE, 2017: 77-85.

[7] LI T Y, GOYAL P, GIRSHICK R, et al. Focal loss for dense object detection [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2020, 42(2): 318-327.

[8] QI C R, YI L, SU Hao, et al. Pointnet++: Deep hierarchical feature learning on point sets in a metric space [C]//Advances in Neural Information Processing Systems. Long Beach: NIPS Foundation, 2017: 5099-5108.

[9] GIRSHICK R. Fast RCNN [C]// Proceedings of the IEEE International Conference on Computer Vision. Washington: IEEE, 2015: 1440-1448.

[10] KU J, MOZIFIFIAN M, LEE J, et al. Joint 3d proposal generation and object detection from view aggregation [J]. CoRR, abs/1712.02294, 2017.

[11] QI C R, LIU Wei, WU Chenxia, et al. Frustum pointnets for 3d object detection from RGB-D data [J]. CoRR, abs/1711.08488, 2017.

[12] ZHOU Yin, TUZEL O. Voxelnet: End-to-end learning for point cloud based 3d object detection [C]// Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City, UT, USA: IEEE, 2018: 4490-4499.

[13] YAN Yan, MAO Yuxing, LI Bo. Second: Sparsely embedded convolutional detection [J]. Sensors (Basel Switzerland), 2018, 18(10): 3337.

[14] JIANG B, LUO R, MAO J, et al. Acquisition of localization confifidence for accurate object detection [C]// Proceedings of the European Conference on Computer Vision (ECCV). Munich, Germany: IEEE, 2018: 784-799.