软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2020,31(3):726−747 [doi: 10.13328/j.cnki.jos.005907]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

基于相关性分析的工业时序数据异常检测*

丁小欧, 于晟健, 王沐贤, 王宏志, 高 宏, 杨东华

(哈尔滨工业大学 计算机科学与技术学院,黑龙江 哈尔滨 150001)

通讯作者: 王宏志, E-mail: wangzh@hit.edu.cn

摘 要: 多维时间序列上的异常检测,是时态数据分析的重要研究问题之一.近年来,工业互联网中传感器设备采集并积累了大量工业时间序列数据,这些数据具有模式多样、工况多变的特性,给异常检测方法的效率、效果和可靠性均提出更高要求.序列间相互影响、关联,其隐藏的相关性信息可以用于识别、解释异常问题.基于此,提出一种基于序列相关性分析的多维时间序列异常检测方法.首先对多维时间序列进行分段、标准化计算,得到相关性矩阵,提取量化的相关关系;然后建立了时序相关图模型,通过在时序相关图上的相关性强度划分时间序列团,进行时间序列团内、团间以及单维的异常检测.在真实的工业设备传感器数据集上进行了大量实验,实验结果验证了该方法在高维时序数据的异常检测任务上的有效性.通过对比实验,验证了该方法从性能上优于基于统计和基于机器学习模型的基准算法.该研究通过对高维时序数据相关性知识的挖掘,既节约了计算成本,又实现了对复杂模式的异常数据的精准识别.

关键词: 异常检测;多维时间序列;时序数据分析;工业大数据;机器学习

中图法分类号: TP18

中文引用格式: 丁小欧,于晟健,王沐贤,王宏志,高宏,杨东华.基于相关性分析的工业时序数据异常检测.软件学报,2020,31(3):726−747. http://www.jos.org.cn/1000-9825/5907.htm

英文引用格式: Ding XO, Yu SJ, Wang MX, Wang HZ, Gao H, Yang DH. Anomaly detection on industrial time series based on correlation analysis. Ruan Jian Xue Bao/Journal of Software, 2020,31(3):726−747 (in Chinese). http://www.jos.org.cn/1000-9825/5907.htm

Anomaly Detection on Industrial Time Series Based on Correlation Analysis

DING Xiao-Ou, YU Sheng-Jian, WANG Mu-Xian, WANG Hong-Zhi, GAO Hong, YANG Dong-Hua

(School of Computer Science and Technology, Harbin Institute of Technology, Harbin 150001, China)

Abstract: Anomaly detection on multi-dimensional time series is an important research problem in temporal data analysis. In recent years, large-scale industrial time series data have been collected and accumulated by equipment sensors from Industrial Internet of Things (IIoT). These data show the feature of diversity data patterns and workflows, which requires high performance of anomaly detection methods in efficiency, effectiveness, and reliability. Besides, there exists latent correlation between sequences from different dimensions. The correlation information can be used to identify and explain anomalies in data. Based on this, this study proposes a correlation analysis based anomaly detection on multi-dimensional time series data. It first computes correlation values among sequences after standardization steps, and a time series correlation graph model is constructed. Time series cliques are constructed according to correlation degree in the

* 基金项目: 国家重点研发计划(2016YFB100703); 国家自然科学基金(U1509216, U1866602, 61602129); CCF-华为数据库创新研究计划(CCF-Huawei DBIR2019005B)

Foundation item: National Key Research and Development Program of China (2016YFB100703); National Natural Science Foundation of China (U1509216, U1866602, 61602129); CCF-Huawei Database System Innovation Research Plan (CCF-Huawei DBIR2019005B)

本文由人工智能赋能的数据管理、分析与系统专刊特约编辑李战怀教授、于戈教授和杨晓春教授推荐.

收稿时间: 2019-07-20; 修改时间: 2019-09-10; 采用时间: 2019-11-25; jos 在线出版时间: 2020-01-10

CNKI 网络优先出版: 2020-01-10 13:34:43, http://kns.cnki.net/kcms/detail/11.2560.TP.20200110.1334.008.html

time series correlation graph. Anomaly detection is processed within and out of a clique. Experimental results on a real industrial sensor data set show that the proposed method is effective in anomaly detection tasks in high dimensional time series data. Through contrast experiments, the proposed method is verified to have a better performance than both the statistic-based and the machine learning-based baseline methods. Research in this study achieves reliable correlation knowledge mining between time series, which not only saves time costs, but also identifies abnormal patterns form complex conditions.

Key words: anomaly detection; multi-dimensional time series; temporal data analysis; industrial big data; machine learning

数十年来,随着工业化和现代化进程的推进,我国制造业持续快速发展.智能制造作为工业大数据的重要应用场景,既是数据的载体和产生来源,也是工业大数据形成的数据产品最终的应用场景和目标[1−3].随着制造业数字化的快速发展,现代化制造生产线、智能产品通过传感器、控制器、智能仪表等[4],实现了对生产运行状态和运行环境的实时记录和感知,已经积累并正在产生大量的工业时序数据.通过对基于采集时间点的多维时间序列数据的分析和挖掘,能够得到宝贵的领域知识,能够对系统的运行状态实现控制、分析、决策和规划[5],以及对被监测到的故障问题和被预测计算到的隐患问题进行诊断、预警、处置、修复.上述过程形成了有效的工业知识产生、提取、应用的积极循环,进而实现了对工业大数据的智能分析.

由于制造系统中存在产品质量缺陷、设备故障、性能下降、外部环境变化等异常问题[6,7],异常工况检测、故障监测、设备健康状态分析等是实现精益生产和智能制造的重要的具体任务,也是工业大数据分析中的重要研究问题[5].如果工业产生中的异常、故障、危机情况不能及时地被有效识别,将导致生产环境存在隐性安全隐患,很可能给整个制造体系带来连带的损失.目前,在高维时间序列数据中的复杂异常状况也逐渐引起重视,一处异常情况的发生往往与多维序列相互影响、产生作用,这些异常模式更难以检测识别.而工业大数据具有大体量、多源异构性强、连续采样、价值密度低、动态性强等特点[1,3,4],这为工业大数据的异常故障检测问题带来了难度和挑战.通过调研,工业时间序列异常检测的研究难点总结如下.

(1) 工业机器设备传感器所采集的时间序列记录了模式多样、多变的工况数据,其数据特点导致了传统时间序列异常检测模型不能很好地适用于工业时序数据异常检测;

(2) 工业时序数据与行业领域联系紧密,难以建立通用的理论计算模型框架,不同类型的工况序列数据模式缺少合理建模方法,时间序列中的值异常、区间异常、模式异常等问题难以有效地描述和量化评价,异常问题容易被误判;

(3) 工业数据采集平台往往包含各类模块化协同工作的传感器设备组,同组的传感器所采集的数据可能模式相似,同时,不同组传感器之间的数据可能存在模式相关关系.这种多维时间序列的相关关系挖掘存在难度,导致存在组内、组间的异常数据容易被漏判、错判;

(4) 已有的大部分多维时间序列的异常检测方法通常时间开销大,方法难以在准确性和效率之间找到有效的平衡,难以满足工业大数据分析对异常检测精度、效率以及可靠度的要求.

而纵观目前已有的时间序列异常检测方法,大部分都专注于解决单维度有周期性或者简单模式时间序列上的异常识别,难以对异常模式和正常模式进行有效的建模区分,不能满足模式多样、工况多变的工业时序数据上的异常监测与检测的需求.此外,由于多维时间序列之间存在一定的相关关系,有助于提高异常检测任务的判别准确性,但复杂的相关性关系难以有效地被建模计算.虽然已有一些基于机器学习的模型可对高维时序数据进行处理,但其计算通常缺乏可解释性,且其计算结果的可靠性难以满足工业时序数据异常检测的需求.已有工作未能实现对多维序列的相关性信息的挖掘和利用,大量错判、漏判的情况导致异常检测方法性能的降低.

基于此,根据真实的制造业数据分析背景,本文研究了基于相关性分析的智能化时序数据异常检测方法.本文的主要贡献总结如下.

(1) 提出了多维时间序列相关性计算方法,并建立时间序列相关图模型,实现对工业时序数据相关机理的深入挖掘,实现了从工业大数据中提取信息进行知识推理.该模型有助于工业知识被高效、自发地提取和应用;

(2) 提出了基于相关性分析的多维时间序列数据异常检测方法,在训练过程中,有效挖掘各个维度上序列

的相关关系,实现了对异常数据的精准定位和识别,从而提高了异常检测方法的准确度和效率,实现了对于模式多样、工况多变的工业时序数据的智能化、全面化的异常检测;

(3) 通过在真实的工业时序数据集上的开展实验,本文验证了所提出方法的有效性和高效性.通过与基于统计和基于机器学习模型的两种基准算法进行对比实验,本文所提出的异常检测方法在准确率、召回率上均优于基准算法,并且有效地节约了计算时间.

本文第 1 节介绍相关研究综述.第 2 节介绍研究问题的预备知识和基本概念,并简述方法框架.第 3 节阐述多维时间序列相关性计算方法,并集合例子分析介绍时序相关性图模型.第 4 节分析基于时序相关性图模型的异常检测算法及案例分析.第 5 节为实验分析,验证算法的可行性和准确性.第 6 节全文总结和研究展望.

1 相关研究综述

近年来,异常检测(anomaly detection)问题在各个学科领域和应用中得到了广泛的研究,其研究目标是找到数据中不满足常态、约束、规则、给定模型的不寻常数据值或模式[8,9],主要应用于网络入侵检测、欺诈行为检测、工业损伤检测、文本异常检测、传感器故障检测等.文献[8]详细地概括了异常模式的特点及异常检测任务的研究难点,介绍了基于分类、基于聚类、基于统计、基于信息论等重要的异常检测方法原理和计算复杂性.

随着数据的时态性和时效性引起重视,研究人员展开了时态数据上的异常检测.Gupta 等人在文献[10]中介绍了各类时态数据:时间序列、数据流、时空数据、时序网络数据上的异常检测问题、研究方法及应用.在时间序列异常检测的研究中,对于检测对象而言,可分为点异常、子序列异常、模式异常这 3 类研究任务.对于异常检测方法而言,主要有基于统计模型[11](如 ARIMA,GARCH 等)、基于聚类[12,13](如 *k*-means,EM,SVM 模型等)、基于相似性度量[14]、基于约束规则[15]等.基于统计的方法通常已知序列的分布,通过维护滑动窗口,计算统计特征指标,实现对异常部位的检测.该方法适用于检测序列中的离散、突变的值异常情况,对于持续的异常序列区间难以有效地识别.基于聚类的方法量化异常点和正常点簇之间的距离来判断离群点,不同聚类模型之间的计算复杂性不同,且检测结果较为依赖聚类的质量.基于相似性度量的方法通过计算经标准化后的序列之间的相似性,来判断是否存有异常数据,但此方法时间开销较大.基于规则约束的方法中,研究人员提出了顺序依赖(sequential dependencies)[16]、速度约束(speed constraints)[17],能够有效利用时间序列中的时序特征对高度异常的数据进行修复,但此方法通常难以满足模式多变的序列异常检测的需求.

对于多维时序数据的异常模式挖掘与检测研究也有了一些进展.在 2006 年,文献[18]分析了高维数据上异常检测问题的任务、方法及其性能效果.文献[19]提出一种在多维数据流中增量式地发现数据相关性和隐藏的变量值,从而将异常数据从正常模式中挖掘出来.文献[20]提出了一种基于距离的高维数据集上异常值检测算法.文献[21]利用了评分向量的概念对异常部位进行概率计算.目前,文献[22]在已有工作基础上,提出了潜在序列相关性计算模型来用于工业数据序列的异常检测.文献[23]研究了时间序列上的异常修复,根据最小修复原则,迭代地对异常序列部位进行更准确地修复.相关工作也促进了时间序列数据清洗的研究,结合应用场景,哈尔滨工业大学团队设计开发了一种综合性、智能化的工业时间序列数据清洗系统 Cleanits[24],能够对多维时间序列的异常问题进行有效识别和修复.

目前,有许多应用案例表明,基于统计和机器学习模型的异常检测方法能较为有效地应用于各类检测任务,可大致分为有监督检测[25](如贝叶斯分类、决策树模型等)、无监督判别(如聚类算法)、无监督参数估计[26−28](如隐马尔可夫模型)等.如何建立序列状态模型,实现对异常序列模式和正常序列的有效区分,一直是该问题的研究重点也是研究难点[9,10].而在工业大数据分析背景上,带标签的数据(即可作为训练集的数据)规模有限,标注所需人力、物力成本高昂[6,8],因此,基于有限标签的弱监督学习[29]方法也是研究方向之一.

虽然文献中提出了许多类型的异常检测方法,但对于模式多变的单维时间序列,异常数据和异常模式仍然难以准确发现.在多维时间序列分析研究中,虽已有一些模型提出,但没有很好地实现对序列间相关性机理的深度挖掘.此外,利用序列相关性信息的异常检测策略尚未建立,缺少高效、可靠、可解释性的智能化异常检测方法.因此,本文的工作补充了利用高维时序数据相关性指导异常检测方法的研究空白,所提出的相关性计算模型

和基于时序相关性图的异常检测模型,也对工业时序大数据分析具有参考价值.

2 研究问题介绍

2.1 基本定义

定义 1(时间序列). 时间序列是由传感器采样和捕获的一系列连续的数据点.一条长度为 N 的时间序列表示为 $S=\langle s_1,s_2,\ldots,s_N\rangle$,其中每个序列点表示的二元组 $s_i=\langle x_i,t_i\rangle$,$x_i$ 是一个实数值,t_i 是时间记录点.对于任意的整数 i 和 j,若 $i<j$,则有 $t_i<t_j$. $T=\{t_i\}_{i=1}^{N}$ 记作时间序列的时间点集合.

定义 2(多维时间序列). $\boldsymbol{S}$ 是一个包含 K 条具有相同时间点集合 T 的时间序列集合,记为 $\boldsymbol{S}=\{S_1,S_2,\ldots,S_K\}$.$\boldsymbol{S}$ 称为 K 维时间序列.

我们根据文献[22]对于工业传感器设备采集数据的介绍,在图 1 中介绍了工业时间序列组数据,见定义 3.

定义 3(工业时间序列组). 令 $\varepsilon=\{E_1,E_2,\ldots,E_M\}$表示设备组(比如一批相同类型的引风机设备),其中,E_m (m=1,…,M)代表第 m 件设备.每件设备 E_m 对应一个传感器组 $\boldsymbol{S}_m=\{S_1^m,S_2^m,\ldots,S_K^m\}$,用于记录不同组件传感器上的数据,其中,$S_k^m$ (k=1,…,K)表示设备 E_m(传感器组 $\boldsymbol{S}_m$)中的第 k 个传感器,即第 k 个时间序列.

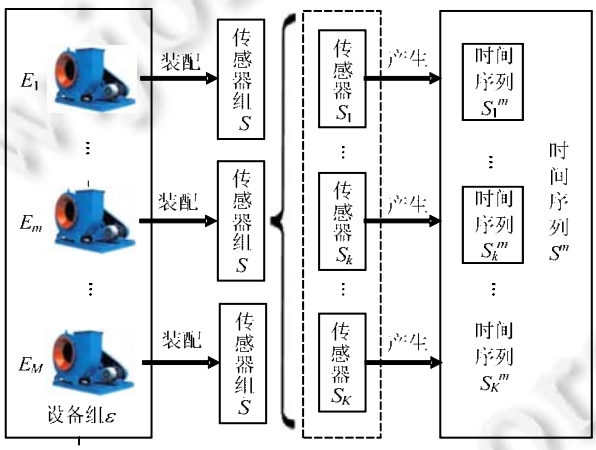


Fig.1 Industrial time series data set

图 1 工业时间序列组

定义 4(序列时间段). 时间区间 $T_{[l:n]}$是设备 E_m 的一个完整的工作时间段,由时间点 t_l 开始,至时间点 t_n 结束,且 $T_{[l:n]}\in T$.

定义 5(时间子序列组). 传感器组 $\boldsymbol{S}_m$ 在时间区间 $T_{[l:n]}$上的全部序列数据称为一个时间子序列组,包含 K 条具有相同时间起止点的子序列.

组成一条原始时间序列的序列时间段间无交集且包含所有时间序列时间点.根据定义 3,我们在图 2 中展示了以 DP105 为例的 4 个传感器上的 4 条时间序列.在处理这段数据时,我们可将整体序列划分为以 250 点为时间段的时间子序列组.序列时间段是我们使用的算法中的基本观察单位.

在异常检测方面,以时间子序列组作为观察的基本单位有两个优点:1) 原始时间序列源源不断被采集记录,时间跨度过大,而序列时间段的长度适中,有助于高效地对序列模式进行建模,可靠性高;2) 由于时间序列反映了设备的工作状态,时间子序列能够反映一段时间内设备部件组的工作状态,因此时间子序列组适合作为最小分析单元.此外,本文的任务主要是检测多维时序数据中,具有一定长度和规模的异常片段,因此本文方法采用时间间隔(序列时间段)作为基本单位,而不是仅仅考虑发生故障的离散时间点.

本文提出的方法在进行相关性计算时,考虑相关性的关系强度和方向.我们用正负号分别代表相关性的正

相关和负相关,用相关性参数绝对值表示相关关系的强度.为了简化分类讨论的情况,我们在定义 6 中总结了相关性关系强度的分类.本文方法着重考虑序列间相关性的强弱,即计算结果值的大小.为了便于计算和理解,本文方法分开记录相关性计算结果的值和正负符号,所定义的相关性计算函数 $Corr(S_i,S_j)$的值域为[–1,1].在后续的计算步骤中,我们设计步骤对相关性的正负方向和强度进行分析,保证计算结果的准确性.

定义 6(序列相关性). 对于给定的具有相同时间点集合的两条时间序列 S_i 和 S_j,$Corr(S_i,S_j)$是定义在 S_i 和 S_j 上的相关性计算函数,序列 S_i 和 S_j 之间的相关关系判定如下.

(1) 若$|Corr(S_i,S_j)|\in[\theta_{high},1]$,则序列 S_i 和 S_j 强相关;

(2) 若$|Corr(S_i,S_j)|\in[\theta_{low},\theta_{high})$,则序列 S_i 和 S_j 弱相关;

(3) 若$|Corr(S_i,S_j)|\in[0,\theta_{low})$,则序列 S_i 和 S_j 无相关关系.

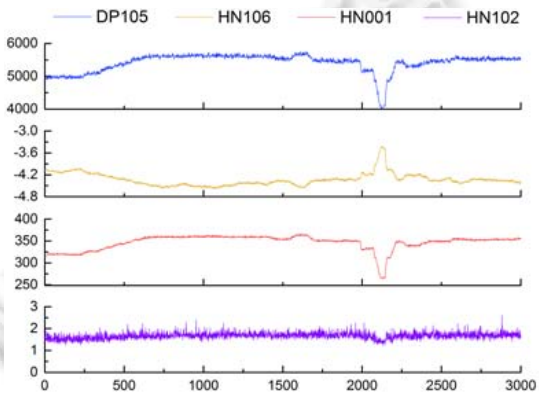


Fig.2 Time series (*x*,*y*): (collection time,data record value)

图 2 时间序列示例

本文将每个传感器组 $\boldsymbol{S}_m$ 的多维时间序列数据作为最小的分析单元进行研究,即:对于一个传感器组内所有维度上的序列进行相关性建模计算,不同设备上的时间序列独立计算.对于一段较长时间内的工业时间序列,同一个传感器组在同一工作周期模式下,序列之间往往具有稳定不变相关性关系,而工作模式的转变可能导致序列间的变相关关系.本文主要研究具有稳定、不变相关性的时间序列,即假定给定的序列之间相关性阈值无较大浮动区间.而对于变相关关系的序列,可以根据其具体工作模式,将其分割为若干个序列时间段.对段内的时间序列,即可利用本文方法进行相关性分析.在计算得到序列之间的相关性后,可利用序列间相关性信息,对隐藏的异常数据进行准确识别.我们在图 3 中通过一个示例展示本文的研究问题.

图 3 展示了 3 条具有很强相关性的时间序列的片段,在由红色矩形划分的时间段内,设备 HNQ01 出现异常问题,虽然 HNQ01 序列在该时间段内的数据值看似在正常范围内变化,但其与另两条序列的相关性发生明显改变,由此推断 HNQ01 出现了异常数据.这是由于该传感器组未出现整体故障,而仅是传感器 HNQ01 发生异常问题.HNQ01 的序列数值先是在 360~363 之间小幅波动,而后迅速下降,推断 HNQ01 传感器发生故障,导致数据采集的失准.在实际情况下,这类情况可能发生在多维时间序列之间,因此本文提出的方法对所有维度时间序列相关性的建模,识别相关性发生变动的子序列部位,通过对这类疑似异常子序列进行全局相关性分析,挖掘并识别出真正发生异常的序列数据.我们在定义 7 中概括了本文研究内容的问题定义.

定义 7(基于相关性分析的异常检测问题定义). 对于给定的一个设备 E_m 对应的 K 维传感器组时间序列数据 $\boldsymbol{S}_m=\{S_1^m,S_2^m,...,S_K^m\}$,实现以下任务:

(1) 设计相关性计算函数 $Corr(i,j)$对 K 维时间序列上任意两条时间序列 S_i^m 和 S_j^m 实现相关性计算量化,记为 $R_{S_iS_j}=Corr(S_i^m,S_j^m)$;

(2) 根据任务(1)中的相关性标记,对 $\boldsymbol{S}_m$ 产生的待检测数据集 $\{S_1'^m,S_2'^m,...,S_K'^m\}$ 上的异常数据进行检测,得

到 N 个异常模式二元组:$\langle T_{[l:n]},AD(\boldsymbol{S}_m)\rangle$,其中,$T_{[l:n]}$记录发生异常的序列时间段,起始于时间点 l,结束于点 n;$AD(\boldsymbol{S}_m)$记录该时间段内所有发生异常的序列编号.

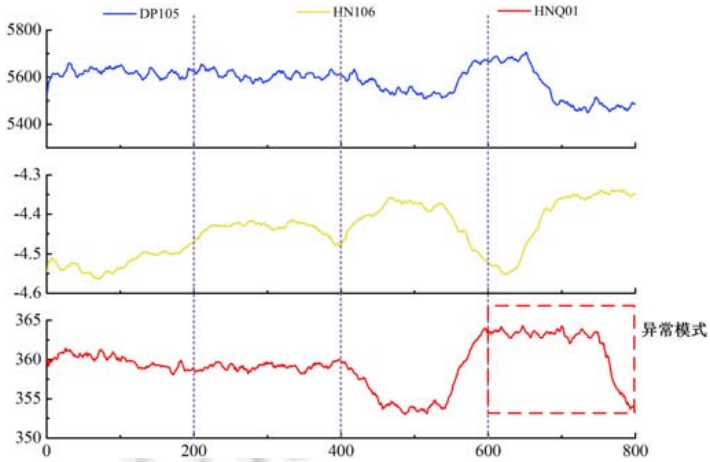


Fig.3 An example of correlation-based abnormal sequence

图 3 基于序列相关性变化的异常示例

2.2 方法概述

经调研发现,偶然发生的异常模式会出现在某一列或者某几列上.我们发现:对于已知相关性关系的多维时间序列,可通过计算其相关性参数是否发生明显变化来筛选出可能出现异常的候选序列集合.但此时仍无法确定具体的异常问题,主要有两个原因:(1) 时间序列的相关性关系具有对称性和无向性,当两条序列的相关性参数发生变化时,不能确定是两者中的哪一条序列发生了异常;(2) 对于一组相关性强的序列集合,其中序列相关性均没有发生变化时,不能判定这些序列没有异常.这可能是由于异常情况同时发生在局部的传感器组,使该传感器组记录的所有序列数据发生相似的异常化改变.因此,我们提出基于时序相关图模型的异常检测方法,首先对序列的相关性进行挖掘分析;在此基础上,提出基于相关性分析的异常检测算法,实现对所有真正的、隐匿性强的异常数据进行有效识别处理.

本文提出的基于相关性计算的多维时间序列异常检测方法如图 4 所示,主要包括数据预处理、时序数据相关性计算以及异常检测 3 个部分.

- 数据预处理:由于采集的原始工业时间序列数据里存在一些数据质量问题[4,5],因此在数据预处理部分,需要对原始的时间序列数据进行时标对齐、缺失值填充等准备性操作,将整理好的高质量数据输入到后面模块进行计算分析;
- 时序数据相关性计算:将准备好的时间序列数据按工作周期模式进行分段,得到若干个时间子序列组,对每个子序列组分别进行序列 PAA 处理(第 3.1 节)、计算相关性参数生成相关性矩阵(第 3.2 节),根据矩阵中的元素值建立时序相关性图,并根据相关性阈值划分图上的时序相关团,该部分具体算法及案例分析在第 3.3 节中介绍;
- 异常检测:在异常检测部分,我们利用已计算得到的时序相关性图模型对待检测数据中隐藏的异常数据进行检测识别.对于彼此存在相关性的序列集合,我们采用基于相关性参数计算异常检测的方法,对时序相关团内以及不同团之间分别进行异常数据挖掘与识别.对于与其他序列无明显相关的序列,我们进行单维序列异常检测.

上述检测步骤能够在节约计算时间的情况下,实现高质量的异常数据挖掘.

按数据处理过程,该方法分为训练阶段和测试阶段.

- 在训练阶段,我们将各传感器上的历史序列数据作为训练数据集,对所有维数上的时间序列进行相关

性计算分析,建立时间序列相关性图模型,并记录各时间序列之间的相关性信息;

- 在测试阶段,我们输入待检测的时间子序列组,利用已训练完成的相关性图模型,对时序相关团内、时序相关团间及孤立点上的序列进行异常模式挖掘与识别,最后判定并输出异常数据在整体测试数据中的具体时间区间和具体维序号,完成异常检测过程.

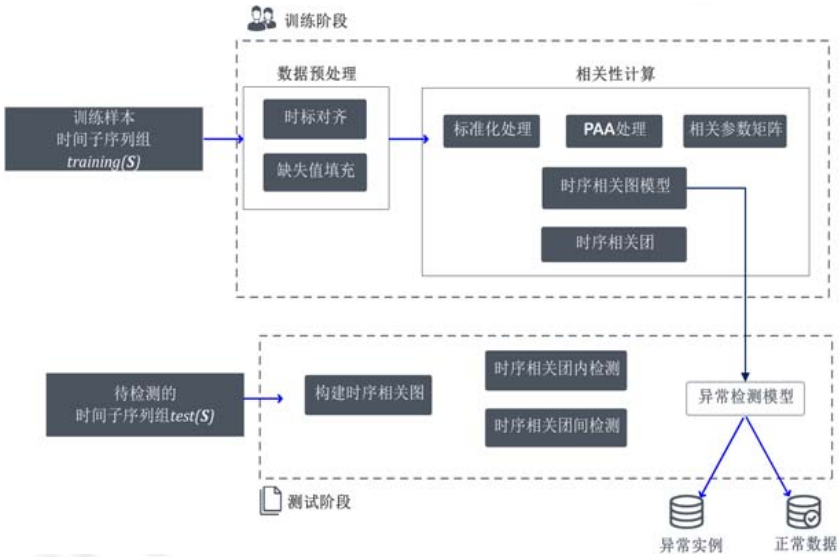


Fig.4 Method framework overview

图 4 本文研究方法框架

3 基于时间子序列组的序列相关性计算

本节对多维时间序列相关性计算过程进行介绍,在第 3.1 节和第 3.2 节分别介绍序列处理和相关性矩阵计算过程,并在第 3.3 节提出时序相关性图模型,进行算法介绍和示例分析.

3.1 序列PAA处理

在对数据进行预处理后,通过时间段划分,我们得到了若干个时间子序列组.对于每一段时间子序列组,每个时间点上的序列取值可看做该时间点上的一个实例.由于时间序列在较短的时间段内具有连续的序列取值变化范围不大的特性,因此我们首先利用逐段聚集平均(piecewise aggregate approximation,简称 PAA)[21]处理每一条序列,进行必要的实例缩减,对序列的浮动特征进行提取,使得后续的计算更为方便、高效和精准.对序列进行 PAA 处理,也是对较长时间跨度内的高维时间序列的主要预处理步骤[12,21].

对时间子序列组进行 PAA 转化前,我们首先将每个时间子序列 S_k^l 标准化(Z-score 标准化)为平均值为 0,标准差为 1 的序列.对长度为 n 的第 k 个时间子序列 $S_k^l=\{s(1)_k^l,...,s(n)_k^l\}$,可以通过 w 维长度向量在 w 维空间($w<n$)中用较短的序列段表示原始序列,记为 $\overline{S_k^l}=\{\overline{s(1)_k^l},...,\overline{s(n)_k^l}\}$,其中,$\overline{S_k^l}$ 的第 i 个元素 $\overline{s(i)_k^l}$ 用公式(1)计算得到:

$$\overline{s(i)_k^l}=\frac{w}{n}\sum_{j=\frac{n}{w}(i-1)+1}^{\frac{n}{w}i}s(j)_k^l \tag{1}$$

PAA 处理将序列的长度从 n 缩减至 w,并将序列分为 w 个相等大小的子段,计算位于框架内的子序列数据的平均值,并且将这些平均值用 w 维矢量表示,同时保留序列的变化特征,算法 1 展示了序列 PAA 处理过程.

算法 1. 序列 PAA 处理.

输入:分段后的一个时间子序列组 $\boldsymbol{S}=\{S_1,S_2,\ldots,S_K\}$;重建后序列长度参数 w;

输出:重建后的时间子序列组 $\boldsymbol{S}'=\{S_1',S_2',...,S_K'\}$.

1. **for** S_k in $\boldsymbol{S}$ **do**
2. $S_k \leftarrow$ ***Z-score***(S_k) /*标准化时间序列*/
3. $n \leftarrow$ ***Len***(S_k) /*时间子序列长度*/
4. **for** i from 0 to w **do**
5. $sum \leftarrow 0$
6. **for** j in ***range***$\left(\frac{n}{w}(i-1)+1,\frac{n}{w}i\right)$ **do**
7. $sum \leftarrow sum+S_k[j]$
8. $data\_reduction[i] \leftarrow sum \times \frac{w}{n}$
9. $\boldsymbol{S}'$.***append***$(data\_reduction)$
10. **return** $\boldsymbol{S}'=\{S_1',...,S_K'\}$

在执行序列 PAA 重建之前,首先需要对各个时间序列进行标准化转换,即执行第 2 行的 *Z-scores* 函数,将每个有量纲的时间序列 S_k^l 标准化为平均值为 0、标准差为 1 的无量纲序列;并且原始序列的数值顺序和变化幅度不会发生改变,便于不同单位或量级的多维数据进行有效的比较处理.在序列标准化后,进行 PAA 处理,将长度为 n 的序列映射至长度为 w 的序列(第 3 行~第 8 行).最后输出经处理后的具有更短长度的时间子序列组 $\boldsymbol{S}'$.

示例分析:我们选取一个传感器组中的 4 条时间序列进行示例介绍,如图 5 所示,从上至下序列名称分别为 DP105,HN106,HNQ01 和 HN102(文中所有序列属性名均已做脱敏处理).左边展示了 2 000 个时间点上原始序列样式,右图是对应的以 w=200 为空间向量进行 PAA 处理后的序列图.可以看出:每条序列经 PAA 处理后变得更为平滑,且长度等比例地由 2 000 缩短至 200;同时,序列的变化趋势信息没有丢失.

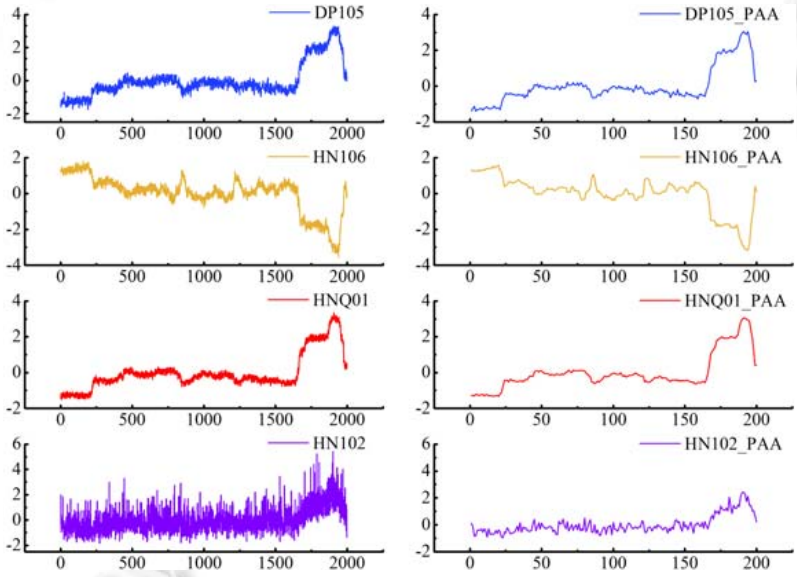


Fig.5 Sequences before and after PAA function

图 5 PAA 处理前后的序列对比图

3.2 序列相关性矩阵计算

对于已重建后的时间子序列组 $\boldsymbol{S}=\{S_1,S_2,\ldots,S_K\}$,需测量时间子序列之间的相关关系,将其表示为时间子序列组的序列相关性.在这一步骤中,使用协方差矩阵(Pearson 系数矩阵[30])初步计算时间子序列组中的序列间的

相关性.在进行 PAA 处理之后,在传感器组 $\boldsymbol{S}$ 的第 l 段(默认长度为 n,下同)时间子序列组中,第 k 个时间子序列表示为 $S_k^l=\{s(1)_k^l,...,s(n)_k^l\}$.在这个序列时间段中,我们在公式(2)中定义相关系数矩阵(series correlation matrix,简称 SCM),用于测量传感器组 $\boldsymbol{S}$ 上第 l 时间段内 K 条序列的相关性,表示为 SCM^l:

$$SCM^l=\begin{pmatrix} R_{11}^l & \cdots & R_{1K}^l \\ \vdots & \ddots & \vdots \\ R_{K1}^l & \cdots & R_{KK}^l \end{pmatrix} \tag{2}$$

其中,元素 R_{ij}^l 表示第 i 个时间序列 S_i^l 和第 j 个时间序列 S_j^l 之间的相关性参数值(series correlation parameter,简称 SCP),其值域为[−1,1].SCM^l 矩阵内的每个元素 R_{ij}^l 取值由公式(3)计算得出:

$$R_{ij}^l=SCP(S_i^l,S_j^l)=\frac{\sum_{m=1}^{n}(s(m)_i^l-\overline{s_i^l})(s(m)_j^l-\overline{s_j^l})}{n-1},\text{其中},\overline{s_i^l}=\frac{\sum_{m=1}^{n}s(m)_i^l}{n},\overline{s_j^l}=\frac{\sum_{m=1}^{n}s(m)_j^l}{n} \tag{3}$$

其中,$s(m)_i^l$ 表示时间序列 S_i^l 在该时间段上第 m 个时间点的数据值,$\overline{s_i^l}$ 和 $\overline{s_j^l}$ 分别为 S_i^l 和 S_j^l 在该时间段内的全部序列数据点的均值.

由此得到了 $\boldsymbol{S}$ 的第 l 时间段的相关系数矩阵 SCM^l.由于在训练阶段,方法需要利用工业上积累的历史数据对序列之间的相关性进行建模计算,且工业设备传感器通常是连续不断地采样[31],因此本文的研究对象——工业时间序列可被看为源源不断到来的流式数据.我们需要将积累的大量历史的序列数据划分成长度合适的时间序列段,再进行后续的计算处理.在训练过程中,我们综合考虑全部 L 个时间段内的相关性计算结果,得到 K 条序列确定的相关性参数值.公式(4)展示了相关系数矩阵 SCM 的计算方法:

$$SCM=\begin{pmatrix} R_{11} & \cdots & R_{1K} \\ \vdots & \ddots & \vdots \\ R_{K1} & \cdots & R_{KK} \end{pmatrix},\text{其中},R_{ij}=\begin{cases}\dfrac{\sum_{l=1}^{L}R_{ij}^l}{L}, & i\neq j\\ 0, & i=j\end{cases} \tag{4}$$

其中,元素 R_{ij} 由全部 L 段时间序列组内的相关性参数值 R_{ij}^l (l=1,…,L)计算均值得出.

示例分析:我们以图 5 中的 4 条序列为示例,介绍相关系数矩阵的计算方法.通过对 DP105,HN106,HNQ01 和 HN102 这 4 条序列使用公式(3)对每个时间点上数据值进行计算,可以得到矩阵 $SCM_{4\times4}$:

$$SCM_{4\times4}=\begin{pmatrix} 0 & -0.95 & 0.98 & 0.58 \\ -0.95 & 0 & -0.95 & -0.57 \\ 0.98 & -0.95 & 0 & 0.59 \\ 0.58 & -0.57 & 0.59 & 0 \end{pmatrix}.$$

根据 $SCM_{4\times4}$ 得知,R_{12}=−0.95,R_{13}=0.98,R_{14}=0.58.矩阵的计算结果和图 5 中的序列模式相互验证,即 DP105 和 HN106 具有较强的负相关性,和 HNQ01 具有较强的正相关性,跟 HN102 具有一定的正相关性.

相关系数矩阵 SCM 是基于统计模型的初步相关性计算结果,我们需要进一步地对序列间的相关关系以及序列组之间的相关关系进行更深入地挖掘和分析.我们在第 3.3 节介绍通过所计算的相关系数矩阵建立的时序相关性图模型.

3.3 时序相关性图模型

在得到 K 条时间序列的相关系数矩阵 SCM 后,为了有效地表示序列间的相关关系,我们提出序列相关性图模型,根据矩阵中元素取值,对序列的相关关系进一步计算.我们首先构建时序相关图,对于一个给定传感器组 $\boldsymbol{S}_m$ 上的 K 维时间序列数据,建立一个无向的时序相关性图 $G_r(\boldsymbol{S})=(V,E)$,其顶点集合记录了所有序列,边集合记录了序列间是否存在大于阈值的相关性信息.具体定义见定义 8.

定义 8(时序相关性图). 对于给定的一个传感器上的 K 维时间序列数据,建立一个无向的时序相关性图,记作 $G_r(\boldsymbol{S})=(V,E)$,将每个序列 S_i^m 记作 G_r 上的一个顶点,组成 K 个顶点的集 $V(G_r)=\{v_i|v_i\in\boldsymbol{S}_m\}$.对于任意两个序列

S_i^m 和 S_j^m,若它们之间的相关性参数值不小于一个给定的相关性阈值 θ_c,则向 S_i^m 和 S_j^m 连一条无向边 $e(i,j)$.因此,G_r 的边集表示为 $E(G_r)=\{e(i,j)|R_{ij}\geqslant\theta_c,R_{ij}\in SCM(\boldsymbol{S}_m)\}$.每条边的边权值表示为 $w(e_{ij})$,且边权值 $w(e_{ij})$记录了 S_i^m 和 S_j^m 间的相关性参数值,即 $w(e_{ij})=R_{ij}(R_{ij}\in SCM(\boldsymbol{S}_m))$.

根据定义 8,我们在算法 2 中介绍构建时序相关性图的流程.在第 1 行,首先初始化无向的时序相关性图 $G_r=(V,E)$;然后我们依次遍历第 3.2 节生成的 $SCM_{K\times K}$ 矩阵的上三角(或等价的下三角),将相关性参数大于等于 θ_c 的两个顶点之间连一条无向边(第 2 行、第 3 行),直至矩阵遍历完成,结束建图.

算法 2. 构建时序相关性图.

输入:相关系数矩阵 $SCM_{K\times K}$,相关性阈值 θ_c;

输出:序列相关图 G_r.

1. 初始化无向图 G_r,包含 K 个顶点
2. **for** i from 1 to K, j from i+1 to K **do**
3. **if** $SCM[i][j]\geqslant\theta_c$ **then** 向 G_r 中加入无向边(v_i,v_j)
4. **return** G_r

示例分析:我们在此举例介绍时序相关性图构建过程,令 $\boldsymbol{S}=\{S_1,S_2,\ldots,S_{14}\}$为时间段 $T_{[l:n]}$上的时序子序列组,根据 $SCM_{14\times14}$,通过算法 2 可以得到时序相关性图 $G_r(\boldsymbol{S})$,如图 6 所示,每条序列作为一个顶点 v 存于图上,其中:v_1,v_4,v_5,v_6 之间两两有边相连;v_7,v_8,v_9,v_{10} 之间两两有边相连;v_{10} 和 v_{12} 相连;序列 2,3,11 因与其他序列相关性较低,因此这 3 个点的度为 0,成为 $G_r(\boldsymbol{S})$上的单点.

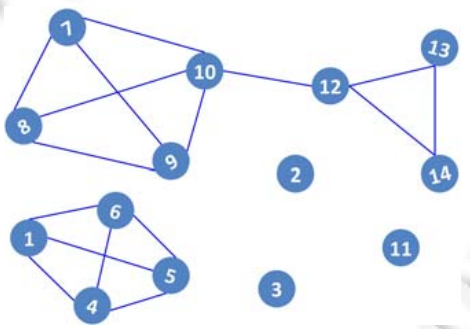


Fig.6 An example of time series correlation graph

图 6 示例:时序相关性图

3.3.1 划分时间序列团

经过算法 2 的计算后,对于传感器组 $\boldsymbol{S}_m$,我们得到如图 6 所示的一个有 K 个顶点、若干条边的时序相关性图 G_r.图上的特点是,具有较强相关性的序列,其顶点间两两相连(如$\{v_1,v_4,v_5,v_6\}$和$\{v_7,v_8,v_9,v_{10}\}$),趋于构成一个局部的时间序列组;具有较弱相关性的时间序列组之间存在少数的边相连,如 v_{10} 和 v_{12};而无明显相关性的序列,相对独立,几乎无边与相连,如 v_2 和 v_3.根据图模型[32]的基本概念,在无向图 G_r 中,如果从顶点 v_i 到顶点 v_j 有路径,则称 v_i 和 v_j 连通.如果图中任意两个顶点之间都连通,则称该图为连通图;否则,称该图为非连通图.G_r 的极大连通子图称为 G_r 的连通分量(connected component),这里所谓的极大是指子图中包含的顶点个数最多.任何连通图的连通分量只有一个,即是其自身;非连通的无向图有多个连通分量.如图 7 所示,该无向图为非连通图中有 3 个连通分量.而图 8 中的无向图为连通图,连通分量即自身.

为了进一步分析图 G_r 上不同相关强度的时间序列组,我们提出时序相关团(time series correlation clique)的概念,通过计算图的连通分量并进行必要的剪枝处理,挖掘和识别多维序列中的异常模式,并且提高异常检测方法的效率.时序相关团的概念如定义 9 所示.

定义 9(时序相关团). 在给定的时序相关图 $G_r(\boldsymbol{S})=(V,E)$上,$C$ 是若干个顶点的集合,表示为 $C=\{v_1,\ldots,v_n\}$,当

C 中元素不小于 1 个时,即 $n\geqslant 2$,若 C 满足以下条件.

(1) $\forall v_i\in C$,有 $v_i\in V(G_r)$;

(2) $\forall v_i\in C$,有 $degree(v_i)\geqslant |C|/2$,即 C 中每个顶点的度大于等于该团内顶点数的一半;

(3) 对于给定的相关性阈值 τ,$\forall v_i,v_j\in C$,有 $w(e_{ij})\geqslant\tau$,$(R_{ij}\in SCM(\boldsymbol{S}_m))$;

(4) C 是 G_r 上满足条件(1)、条件(2)的最大顶点集合,即:不存在 $v_j\in\boldsymbol{S}$ 且 $v_j\notin C$,使得 $C\cup v_j$ 仍满足条件(1)、条件(2).

则称 C 为 G_r 上的一个时序相关团.对于 G_r 上未被划分进任一个时序相关团的孤立点 v,将 v 记作一个仅包含单个元素的单点时序相关团 $C=\{v\}$.

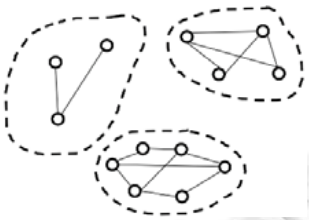


Fig.7 A graph with 3 connected components

图 7 非连通图有 3 个连通分量(虚线)

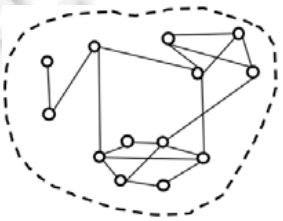


Fig.8 Itself as the connected component

图 8 连通图分量为自身的连通图

根据定义 9,我们将一个时序相关图上 K 条时间序列划分为若干个时序相关团,得到定理 1 对图 G_r 的表示方法.

定理 1. 图 G_r 可表示为若干个时序相关团的集合,即 $G_r=\{C_1,\ldots,C_m\}$,满足:

(1) $\bigcup_{i=1}^{m}V(C_i)=V(G_r)$;

(2) $\bigcap_{i=1}^{m}C_i=\varnothing$.

我们遍历图上每个顶点形成的连通分量集合,根据定义 9 寻找并划分 G_r 上的时序相关团,具体算法记为 $TSCC(G_r)$,见算法 3.

算法 3. 划分时间序列团.

输入:时序相关性图 G_r,相关性阈值 τ;

输出:G_r 上所有时序相关团的集合 $\boldsymbol{C}=\{C_1,\ldots,C_n\}$.

```
1.  初始化数组 prunedNode=V(G_r),表示待形成团的顶点
2.  while True
3.    if prunedNode 为空 then break
4.    初始化数组 visit[V(G_r)],标记所有顶点未访问
5.    初始化集合 C_temp,表示待剪枝的时序相关团
6.    for prunedNode 中的每个顶点 v do
7.      if visit[v]为未访问 then
8.        以 v 为顶点进行深度优先搜索,遍历所有未标记过的顶点
9.        C_i←满足 w(e_ij)≥τ的顶点 v_i 和 v_j,并将 visit[v_i]和 visit[v_j]标记为已访问
10.     else C_i←{v},并将 visit[v]标记为已访问
11.     将顶点集加入集合 C_temp
12.   令 prunedNode 为空
13.   for C_i∈C_temp do
14.     S←算法 4 处理得到的时序相关团和散点
```

15. 将 $\boldsymbol{S}$ 中时序相关团和散点加入分别加入 $\boldsymbol{C}$ 和 *prunedNode* 中

16. **return** $\boldsymbol{C}\{C_1,\ldots,C_n\}$

首先,我们对于图上每个顶点维护一个标记 *visit*[·];然后遍历每一个顶点 v_i,以广度优先搜索找到与 v_i 相连且边权值大于给定阈值的所有顶点,将其加入同一个时序团中(第 7 行~第 9 行).在将所有可能的顶点加入团后,我们需要根据定义 9 对团的满足条件进行判定.我们提出算法 4 实现对团的满足性的判断:若团内每个顶点数的度均满足大于团内顶点数的一半,且团内每一条边权值均大于给定阈值,则该团成立(算法 3 的第 14 行);若存在不满足上述条件的点,则对当前团进行剪枝和删除点的操作,并将从该团内删掉的点返回算法 3 的第 3 行重新进行分析,直至找到图上所有满足条件的时序相关团 $\boldsymbol{C}=\{C_1,\ldots,C_n\}$,作为结果返回,算法 3 结束.

算法 4. 时序相关团剪枝算法.

输入:一个时序相关团 C_i;

输出:剪枝处理后的新时序相关团 C_i' 和散点的集合 $\boldsymbol{S}=\{C_i',node_1,\ldots,node_n\}$.

1. 初始化集合 $\boldsymbol{S}$
2. **while** True
3. 选取 C_i 中顶点度最小的顶点 $node_{\min}$
4. **if** $degree(node_{\min})<|C_i|/2$ **then** 将 $node_{\min}$ 从 C_i 删掉,并加入 $\boldsymbol{S}$
5. **else break**
6. **return** $\boldsymbol{S}=\{C_i',node_1,\ldots,node_n\}$

示例分析:我们在图 6 的基础上介绍时序相关团的寻找和划分过程.由顶点 v_1 开始进行广度优先搜索,发现其与 v_4,v_5,v_6 形成一个时序相关团,满足定义 9,则将其记为团 A:$C_A=\{v_1,v_4,v_5,v_6\}$.类似易得团$\{v_7,v_8,v_9,v_{10}\}$和团$\{v_{12},v_{13},v_{14}\}$.划分完成后,将剩余的单点直接变为一个团.将团的标记按团内最小的顶点编号的升序进行排序得:$C_B=\{v_2\},C_C=\{v_3\},C_D=\{v_7,v_8,v_9,v_{10}\},C_E=\{v_{11}\},C_F=\{v_{12},v_{13},v_{14}\}$.时序相关团的判定结果如图 9 所示.

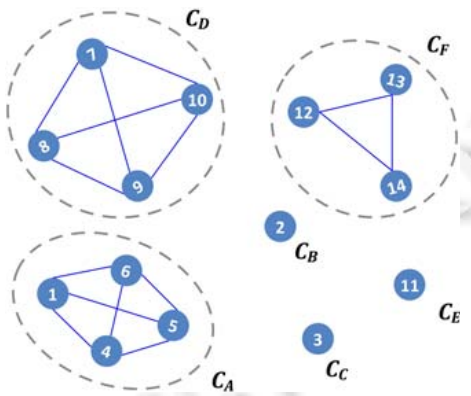


Fig.9 An example of determining time series correlation cliques

图 9 示例:划分时序相关性团

3.3.2 团间相关性计算

在得到时序相关团后,我们需要对时序相关团之间的相关关系强度进行分析,主要分为时序相关团内分析和时序相关团间分析.团内的相关性分析可以由团内各个顶点之间的边权值计算得到,而不同时序相关团之间的分析需要进一步处理.

对于 G_r 上的一个时序相关团 C_i,在计算 C_i 与图中其他团之间的相关关系强度,若逐一选取 C_i 中所有顶点进行计算,会使计算结果空间和计算时间的开销都很大.由于每个时序相关团内顶点间的边权值都很高,即同个团内时间序列均具有显著可靠的相关性,我们可以选取团上的特征序列代表该团进行计算,实现在相关性信息不丢失情况下降低计算规模.在特征序列的选择策略上,我们有两种可供选择的方案.

- 方案 1:将时序相关团内的所有时间序列归一化后,计算均值得到的新时间序列作为特征序列;
- 方案 2:选取时序相关团内与其他时间序列相关性最强的时间序列作为代表序列.

考虑到方案 1 在训练过程中,各序列模式被迫平均到均值序列上,当训练数据集规模增加时,各个时序相关团的特征序列极易趋于相似,难以得到可靠的训练结果;另一方面,方案 1 的均值序列并非真实序列,缺少实际代表性意义.此外,本文通过大量实验验证了方案 1 难以得到高质量的训练效果.因此,本文选择方案 2 进行计算,时序相关团的特征序列在定义 10 中给出.

定义 10(特征序列). $C=\{v_1,\ldots,v_n\}$是时序相关性图 G_r 上已知的一个时序相关团,当 C 中元素不小于 1 个时,存在一个 $v_i\in C$,满足 $v_i=\arg\max\sum_{j=1}^{n}w(e_{ij})$,即,点 v_i 表示的序列 S_i 是与时序相关团 C 中其他序列相关性参数值之和最大的序列.v_i 记为时序相关团 C 的特征点,即 $v^*(C)=v_i$,v_i 表示的序列 S_i 记为 C 的特征序列,即 $S^*(C)=S_i$.当 $C=\{v\}$是单点时序相关团时,其唯一元素 v 记作 C 的特征点,v 表示的序列记作 C 的特征序列.

由于本文在实际的相关性参数 R_{ij} 计算中采用浮点运算,因此对于一个给定的时序相关图 C,我们总能找到唯一特征点 $v^*(C)$.证明过程不再赘述.

在特征序列选取完成后,我们对时序相关团间的相关关系进行分析,对 G_r 上的任意两个时序相关团 C_i 和 C_j,分别计算得到特征序列 $v^*(C_i)$和 $v^*(C_j)$.利用第 3.2 节的方法建立相关性参数矩阵 $SCM(v^*)$,得到矩阵中元素 $R_{ij}=SCP(v^*(C_i),v^*(C_j))$,其中,$C_i,C_j\in G_r$.$R_{ij}$ 记录了 C_i 和 C_j 的相关性关系,我们根据 R_{ij} 的取值对时序相关团的相关关系进行分类标记,见表 1.

Table 1 Relation classification between series correlation cliques

表 1 时序相关图相关关系分类

$SCP(v^*(C_i),v^*(C_j))$	[−1,−0.7]	(−0.7,0.4]	(−0.4,0.4)	[0.4,0.7)	[0.7,1]
R_{ij}	负强相关	负弱相关	不相关	正弱相关	正强相关

对于两个时序相关团 C_i 和 C_j,按照相关性强度,分为强相关和弱相关;按照相关性方向,分为正相关和负相关.对于 $R_{ij}\in(-0.4,0.4)$,我们认为 C_i 和 C_j 无明显相关关系,在本文模型中记作不相关,在后续异常检测过程中,无需进行基于相关性的计算.算法 5 展示了时序相关团间关系的计算和标记过程.

算法 5. 标记时间序列团间相关关系.

输入:图 G_r 上已划分好的时序相关团集合 $\boldsymbol{C}\{C_1,\ldots,C_n\}$;

输出:被标记相关关系 R_{ij} 的时序相关图模型 G_r^R .

1. 计算 $\boldsymbol{C}$ 上所有团的特征序列,得到特征序列集合 $V^*=\{v^*(C_1),\ldots,v^*(C_n)\}$
2. $SCM_{n\times n}\leftarrow SCP(V^*)$
3. 初始化无向图 G_r^R ,包含 N 个顶点
4. **for** i from 1 to N, j from i+1 to N **do**
5. **if** $R_{ij}\neq$不相关 **then**
6. 向 G_r^R 中加入无向边(v_i,v_j),根据表 1 标记其关系 R_{ij}
7. **return** G_r^R

我们首先根据定义 9 找到所有时序相关团的特征序列,记录并维护一个特征序列集合 V^*.在 V^*上,我们执行第 3.2 节提出的 SCP 函数,计算得到 $n\times n$ 的相关系数矩阵 $SCM_{n\times n}$(第 1 行、第 2 行).然后在第 3 行初始化一个无向图 G_r^R ,遍历 V^*所代表的所有时序相关团,将任意两个团 C_i 和 C_j 以权值为 R_{ij} 的边连接,并根据表 1 标记其关系分类.最后得到了完整的时序相关图结构,其中包含时序相关团集合和团间相关性参数值和相关关系类型.

示例分析:在得到图 9 上的时序相关团划分后,我们执行算法 5,对团上的相关性进行标记.如图 10 所示,已知 G_r^R 上已划分得到 6 个时序相关团:$C_A=\{v_1,v_4,v_5,v_6\}$,$C_B=\{v_2\}$,$C_C=\{v_3\}$,$C_D=\{v_7,v_8,v_9,v_{10}\}$,$C_E=\{v_{11}\}$,$C_F=\{v_{12},v_{13},v_{14}\}$.对 6 个时序相关团分别计算得到特征序列 $V^*=\{v_5,v_2,v_3,v_9,v_{11},v_{12}\}$.对 6 条特征序列计算相关参数矩阵

$SCM_{6\times6}$.经计算, $R_{C_AC_B}<-0.7, R_{C_BC_E}\in[-0.7,-0.4]$,则分别将$(C_A,C_B)$和$(C_B,C_E)$之间标记为强负相关、弱负相关.类似地,得到$(C_A,C_E)$,$(C_A,C_C)$,$(C_C,C_D)$,$(C_D,C_F)$之间是弱正相关.我们在图 10 中展示已标注的时序相关团,其中,红色虚线表示弱正相关,蓝色虚线表示弱负相关,蓝色实线表示强负相关.至此,完成了时间序列组 $\boldsymbol{S}=\{S_1,S_2,\ldots,S_{14}\}$的时序相关图模型的构建.

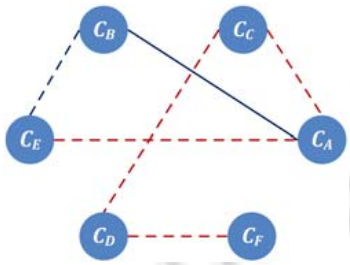


Fig.10 An example of labelling time series correlation cliques

图 10 示例:标记时序相关性团的相关性

4 基于时间序列图模型的异常检测

4.1 算法介绍

在构建完成时序相关图模型后,我们接下来利用该模型对多维时间序列异常模式进行检测.在工业时序数据上,异常模式往往以较低概率出现在单维或者多维序列上,且异常模式会持续一段时间,而不是出现在少量离散的时间点上.由于持续一定时长的异常模式的检测需求比稀疏的离散异常点检测更有重要性,本文的异常检测任务主要是发现并识别那些持续一定时长的异常问题数据.

在检测过程中,我们仍然采用时间子序列组作为基本分析单位,在测试过程中,逐段进行异常检测分析.对于一个传感器组 $\boldsymbol{S}$ 的第 l 个时间段的 K 维时间序列,检测过程见算法 6.

算法 6. 多维时间序列异常检测算法.

输入:时间子序列组 $\boldsymbol{S}=\{S_1,S_2,\ldots,S_K\}$经算法 5 得到的时序相关图模型 G_r^R ,相关性阈值θ_c;

输出:异常序列集合 $AD(\boldsymbol{S})=\{S_1,S_2,\ldots,S_{K'}\}$.

1. 初始化异常序列集合 $AD(\boldsymbol{S})$为空
2. 初始化无向图 G_A
3. **for** $C_i\in\boldsymbol{C}(G_r^R)$ **do**
4. **if** C_i 的度为 0 and $\boldsymbol{len}(C_i)\leqslant2$ **then**
5. 将 C_i 中序列做单维时间序列进行异常检测,若异常则将其加入 $AD(\boldsymbol{S})$
6. **continue**
7. **if** $C_i=\{v\}$是单点时序相关团 **then** 将 v 作为特征序列 $v^*(C_i)$加入 V^*
8. **else** 初始化无向图 G_B,令$|V(G_B)|\leftarrow\boldsymbol{len}(C_i)$ /*时序相关团内异常检测*/
9. **for** $v_x,v_y\in C_i$ and $e(v_x,v_y)\in E(G_B)$ **do**
10. **if** $R_{xy}<\theta_c$ **then** 向 G_B 加入无向边 $e(v_x,v_y)$ /*G_B 记录存在异常的边*/
11. $G_B\leftarrow G_B\backslash G_B$ 中没有度的点 v
12. **if** G_B 是二分图 **then** $AD(\boldsymbol{S})\leftarrow$Hungary 算法求解其最小点覆盖
13. **else** $AD(\boldsymbol{S})\leftarrow$贪心算法求得异常序列
14. $C_i\leftarrow C_i\backslash AD(\boldsymbol{S})$中的点 /*去掉时序相关团内异常的点*/
15. $V^*\leftarrow$当前 C_i 的特征序列 $v^*(C_i)$
16. **for** $C_x,C_y\in\boldsymbol{C}(G_r^R)$ **do** /*时序相关团间异常检测*/

17. **if** $R_{C_xC_y} \neq G_r^R$ 中边(C_x,C_y)的标记 **then** 向 G_A 加入无向边 $e(C_x,C_y)$

18. $G_A \leftarrow G_A \backslash G_A$ 中没有度的团 $\boldsymbol{C}$

19. **if** G_A 是二分图 **then** $AD(\boldsymbol{S})$←Hungary 算法求解其最小点覆盖

20. **else** $AD(\boldsymbol{S})$←贪心算法求得异常序列

21. **return** $AD(\boldsymbol{S})$

算法 6 首先初始化一个异常序列数据集合 $AD(\boldsymbol{S})$,然后进行异常检测,主要分为两个步骤:第 3 行~第 15 行进行时序相关团内的异常序列检测;第 16 行~第 20 行进行时序相关团之间的异常检测.在团内检测过程中,我们依次选取 G_r^R 上的每一个时序相关团 C_i,首先对其特点进行分析,如果团 C_i 是图 G_r^R 上孤立的团,且团内序列数不大于 2,对于这样的情况,序列相关性关系未能对异常检测提供信息,因此算法对该团上序列进行单维异常检测,并将结果记入 $AD(\boldsymbol{S})$(第 4 行、第 5 行).如果 C_i 是一个单点时序相关团,则将其唯一元素 v 作为特征序列加入特征序列集合 V^*.

对于其他时序相关团 C_i,我们维护一个无向图 G_B 记录 C_i 的异常检测操作过程.我们遍历团内所有边,检测边权值的大小.根据时序相关图的定义,同一个团内的序列都是以很高的相关性阈值相连,因此若检测到边权低于给定的相关性阈值θ_c,则认为异常存在于该边相连的两个点之中,我们将出现异常权值的边加入 G_B(第 9 行、第 10 行).此时,G_B 与 C_i 有相同的顶点结构,且 G_B 的边集合记录了候选的异常顶点.在第 10 行,我们去掉 G_B 上没有边相连的顶点,即没有发生异常的顶点.在得到的新 G_B 上,计算确定异常的具体位置.记录候选异常序列的图 G_B 上,一个发生实际异常的序列与其他几条与之强相关的正常序列分别有边相连.由于异常问题往往仅存在于某列或某几列的时间序列上,若序列 S_x 发生异常,则 S_x 与其相关序列的相关性参数均发生异常.即:G_B 上的边 $e(v_x,v_y)$通常记录的是一条异常序列 S_x 和一条正常序列 S_y,并且异常序列 S_x 与其他相关序列也有边相连.此外,S_x 和 S_y 也可能都是异常序列.

根据最小修复代价原则[17,23],图 G_B 上的异常判定问题被转化为无向图的最小顶点覆盖问题.虽然无向图上最小点覆盖问题已被证实是 NP 完全问题[7],但由于设备上传感器的个数有限,需要计算的时序相关图规模不很复杂,因此我们提出方法实现对该问题的高效率求解.通过大量实验分析知,图 G_B 通常是二分图结构,因此在检测时,我们首先对 G_B 进行二分图判定.由于二分图的最小顶点覆盖问题不是 NP 完全问题,因此可实现对计算效率的优化.

根据 König[33]定理,二分图最小点覆盖包含的点数等于二分图最大匹配包含的边数.我们可以对二分图构造出一组点覆盖,其包含的点数等于最大匹配包含的边数.具体流程如下.

- 利用 Hungary 算法[32,34],求得二分图最大匹配;
- 从左部每个未匹配的点出发,执行一次深度优先搜索寻找二分图的增广路,标记访问过的顶点;
- 取出左部未被标记的顶点、右部被标记的点,得到二分图最小点覆盖.

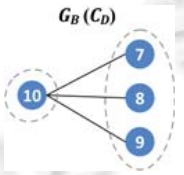
由此,在第 12 行,我们将对 G_B 执行的二分图的最小顶点覆盖结果作为异常序列的识别结果输入 $AD(\boldsymbol{S})$集合.若 G_B 判定为不是二分图,我们采用贪心策略去寻找真正的异常序列,具体步骤是:在 G_B 中选择一个度最大且至少为 1 的顶点 v,将其标注为异常,然后删除与 v 相连的边.重复执行这一操作,直到所有顶点的度均为 0.以此方法求得异常序列集合,将其加入 $AD(\boldsymbol{S})$中.虽然贪心策略不能保证得到的解是无向图上最小点覆盖,但其在本文研究的时序相关图上可以高效地找到发生异常的序列.此外,在后面步骤中(第 14 行、第 15 行),我们去掉异常序列重新计算时序团的特征序列进行迭代的计算,保证加入 $AD(\boldsymbol{S})$结果集合中的序列是真实的异常序列,尽可能地避免错判和漏判.

在时序相关团内计算后,我们进行时序相关团之间的计算,维护一个 G_A 记录团之间的异常部位.与 G_B 结构相似地,G_A 中的顶点记录全部时序相关图集合,且 G_A 的边集合记录了存在异常关系的顶点对.通过对 G_A 的计算,实现对整体发生异常而未能被计算得出的异常团进行识别.其计算流程与时序相关团内计算相似,此处不再赘述.在遍历所有的时序相关团后,算法 6 结束,输出的结果即为检测出的异常序列.

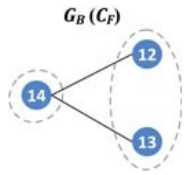
4.2 示例分析

在得到图 10 中时序相关图模型后,我们将其作为训练结果.对于测试集的一段序列组数据 $\boldsymbol{S}=\{S'_1,S'_2,...,S'_{14}\}$,我们对数据进行时序相关图建模,在计算得出测试数据 $\boldsymbol{S}$ 的新时序相关图 $G_r^R(\boldsymbol{S})$ 后,我们执行算法 6,分析数据上是否存在异常序列.我们依次遍历每一个时序相关团,计算发现图上不存在孤立点,对于单点时序相关团 $C_B=\{v_2\}$,$C_C=\{v_3\}$,$C_E=\{v_{11}\}$,我们直接将其中唯一元素加入序列集合 V^*(第 4 行~第 7 行).然后在第 8 行,开始进行团内异常检测.通过相关性阈值θ_c,我们发现 C_D 上的异常边{(7,10),(8,10),(9,10)},则将这 3 条边记录在 $G_B(C_D)$上,如图 11(a)所示.对其进行二分图检测,判定其为二分图,则对 $G_B(C_D)$执行最小顶点覆盖求解(第 11 行、第 12 行),得到 $AD(\boldsymbol{S})=\{v_{10}\}$,即 v_{10} 所代表的序列 S'_{10} 在该时间段内出现异常问题.类似地,对于 $G_B(C_F)$,我们检测到 v_{14} 存在异常.我们去掉团内异常的点,重新计算其相关性参数,满足原相关性关系,则确定地输出检测到的异常点.

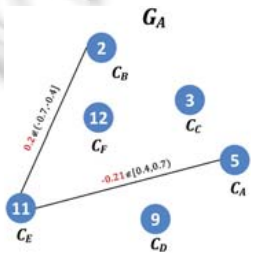
然后进行时序相关团之间的异常检测,用图 G_A 记录异常边.计算发现存在(C_A,C_E)和(C_B,C_E)两条异常边,经二分图最小顶点覆盖计算得 C_E 是异常团.由于 C_E 只包含一个元素,无需再进行团内元素计算,直接将其加入异常集合 $AD(\boldsymbol{S})$中.至此,计算得出该时间段上 $\boldsymbol{S}$ 中存在的异常序列为 $AD(\boldsymbol{S})=\{v_{10},v_{11},v_{14}\}$.



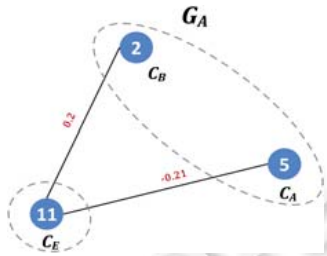
(a) 团 C_D 内异常检测



(b) 团 C_F 内异常检测



(c) 团之间的异常检测



(d) 团之间的异常检测(去除孤立点后)

Fig.11 An example of labelling time series correlation cliques

图 11 示例:标记时序相关团相关性

4.3 算法分析

本文提出的整体检测算法的计算时间消耗主要在时序相关性图构建与计算和基于最小点覆盖问题的异常检测这两个步骤上.

(1) 时序相关图模型构建

在时序相关图模型构建的过程中,我们将训练集分为 N 个时间子序列组,每组有 K 维长为 n 的时间子序列.对这些时间子序列进行相关系数矩阵计算,是我们算法的一个耗时点.我们需要对每两条时间子序列进行相关性计算,每次计算需要统计全部的时间子序列组,消耗的时间复杂度为 $O(NK^2)$.

在判定时序相关图 G_r 上的时序相关团时,记图上有 M 个时序相关团,则判定时间序列团模型通过广度优先遍历且相关系数矩阵相当于邻接矩阵,故时间复杂度为 $O(K+E)$.时间序列团间相关关系计算与相关系数矩阵计算复杂性相同,消耗的时间复杂度为 $O(N\cdot M^2)$.由于时间序列团数量通常小于等于序列维数,即 $M\leqslant K$,因此时序相关图构建的时间复杂度为 $O(N\cdot K^2)$.由于每个设备包括的传感器数量有限(经调研,CPS 系统每个传感器组中包含的传感器数量平均情况小于 100 个),因此每个图结构规模不大,$O(N\cdot K^2)$的复杂度可以满足训练过程的耗时需求.

(2) 基于时序相关图模型的异常检测

在异常检测的过程中,对于待检测的 N 个 K 维时间子序列组,在进行团内和团间计算构建 G_A 和 G_B 的过程中,G_A 上至多有 K 个顶点和 $(K/2)^2$ 条边,因此执行 Hungary 算法对二分图计算的最坏复杂度为 $O(K^3)$,则执行异常检测的时间复杂度为 $O(NK^3)$.但由于序列数量 K 规模不大,二分图计算的时间复杂度很难达到最坏情况,通常每个 G_B 仅包括数个顶点,数量远小于 K,通常可以在 $O(N{\cdot}K{\cdot}m^2)$时间内完成检测,其中,m 代表 G_B 包含的顶点个数,一般是一个小于 10 的正整数.

5 实验分析

5.1 实验设定

(1) 数据集

本文应用国内某火力发电厂的引风机机组数据进行实验.我们将连续 3 个月的历史采样数据作为训练集.该设备每 8s 记录一次数据,一共包括 63 列时间序列数据.经本文提出的预处理方法去除无效或低质量数据后,最终采用 48 列总计 55 万个时间点上数据进行实验.

(2) 对比算法

在实验中,我们实现了上文介绍的所有算法,并将本文提出的算法称为 CGAD 算法(correlation graph model based anomaly detection).为了客观地验证本文方法的性能,我们实现了两种时间序列异常检测方法作为基准算法,进行性能对比实验.

- AR 算法[11]:一种基于统计的序列异常检测的基本算法,维护一个长度为 k 的动态窗口,计算窗口内序列的均值、方差及其他统计性特征值,以此来预测第 k+1 个窗口的实际数据值是否满足算法对其值的预测,若不满足,则视为发生异常.
- LCAD 算法[22]:基于机器学习模型的异常检测算法,首先对多维序列进行协方差矩阵计算,实现对其相关性的测量,并从矩阵中提出单维的特征向量作为序列的特征值,用高斯分布模型训练样本集合,对于待检测数据,用最大期望(EM)算法得到数据正常或异常的概率分类.

(3) 计算指标

我们分类的目标只有两类,即正例(positive)为正常数据,负例(negative)为异常数据.我们将时间子序列组作为一个实例单位,可将实验结果分为以下 4 类,分别是:

- True Positives (TP):实际为正常且算法检测为正常的实例数;
- False positives (FP):实际为异常但算法检测为正常的实例数;
- False negatives (FN):实际为正常但算法检测为异常的实例数;
- True negatives (TN):实际为异常且算法检测为异常实例数.

在对结果进行分类后,我们可通过计算准确率和召回率来评价算法的性能.

- 准确率(P): $Precision = \dfrac{TP}{TP + FP}$;
- 召回率(R): $Recall = \dfrac{TP}{TP + FN}$.

在实验中,我们选取 2 000 个时间点上 48 列数据作为一个时间段,即,将时间长度为 2 000 的数据记为一个数据组.对用于实验的 55 万个时间点上的 48 维时间序列,我们在训练集中最多使用了 200 个数据组,即有 40 万个时间点数据构成训练样本.在测试集中,最多使用了 50 个数据组.异常模式较为均匀地出现在 48 列数据上,我们将一条时间序列上的一个长度大于 400 点的异常模式记为一个异常实例,任意异常实例的长度值域约为[400,1000].异常实例较为分散地存在在待检测数据中.异常实例总数最多约为 1 000 个.

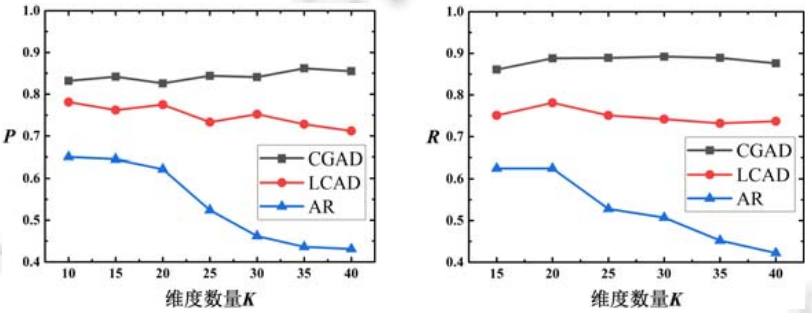
5.2 方法有效性分析

分别测试序列维数总数、异常实例总数、测试集规模和训练集规模对上述 3 种算法检测性能的影响.

(1) 序列维数的影响

实验测试了用 200 组数据做训练集、共有 600 个异常实例、序列的数量从 10 列增长到 40 列情况下,本文算法和两个对比算法的准确率和召回率.从图 12 上可以看出:随着序列数的增加,AR 算法的性能在序列数量大于 20 列之后开始明显下滑.主要有两个原因:(1) 基于统计的 AR 模型本身无法对工业时序数据的特征进行准确的概括和计算,导致对异常数据的识别效果不好;(2) AR 算法只对单维时间序列做检测,而没有考虑到序列之间的相关性,导致许多难以被统计模型检测到的隐藏异常问题未被发现.

随着序列数的增加,本文提出的 CGAD 算法和 LCAD 算法的召回率比较稳定.这一结果反映了多维时间序列相关性的计算能有效提高异常检测性能.CGAD 算法在序列数量增加至 20 列之后,对异常识别的准确率达到 82%,召回率达到 88%.随着序列数量的增加,准确率和召回率均趋于稳定.这验证了 CGAD 算法对时序数据相关性建模的必要性和可靠性,也证实了本文提出的检测算法能准确地识别出序列中隐藏的异常部分.虽然 LCAD 算法的 R 值随着序列数量的增加而保持稳定,但其准确率有所下降,且一直未能达到 80%.这是由于 LCAD 算法虽然考虑了序列间的相关性,提高了检测质量,但在异常检测方法上,本文提出的 CGAD 算法的识别效果要更胜于 LCAD 算法.



(a) 准确率对比 (b) 召回率对比

Fig.12 Varying total number of time series

图 12 序列维数变化对算法性能的影响

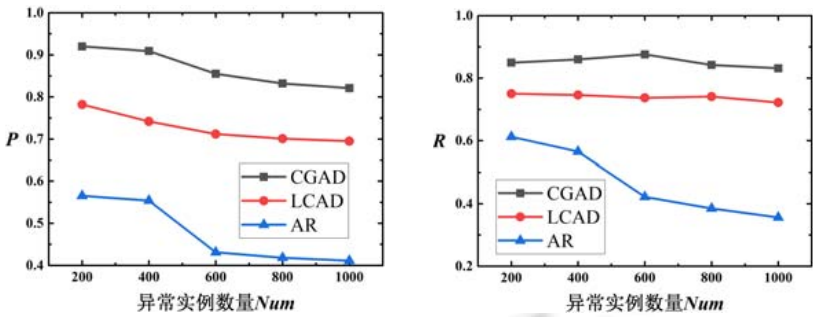
(2) 异常实例总数的影响

图 13 展示了用 200 组数据做训练集,序列维数为 40 列时,异常实例总数的变化对 3 种算法性能的影响.

随着数据中异常数据总数的增加,AR 算法性能存在明显下降.这说明数据中异常值越多,越影响 AR 算法对序列特征提取的可靠性.CGAD 算法和 LCAD 算法的 P 值、R 值高于 AR 算法,但是随着异常实例数量变多,LCAD 算法的准确率有明显的下降,而本文 CGAD 算法的 P 值稍有下滑,但仍然保持在 82%以上,R 值稳定在 83.1%~87.6%之间.这说明本文设计的时序相关团内、团间的异常检测策略,能在多异常的复杂数据条件下,检测效果能保持稳定.

(3) 测试集规模的影响

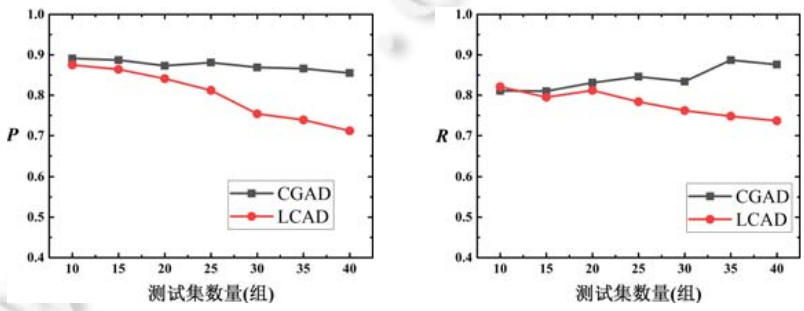
本组实验测试了用 200 组数据做训练集,在 40 列序列上,通过改变测试集规模的大小,分析算法性能的变化趋势.由于 AR 算法没有训练过程,因此这里仅比较 CGAD 算法和 LCAD 算法两种算法的性能.从图 14 看出,测试集数量的增加对 LCAD 算法的性能影响不大,P 值和 R 值在 25 组之前变化不明显,在 25 组之后均有下降.这说明测试集数量的增多会一定程度降低 LCAD 算法性能.值得注意的是:测试集规模的增大时,CGAD 算法性能的 P 值趋于稳定,而且 R 值略有提高.这证实了本文 CGAD 算法对序列间相关性建模的有效性.随着待检测数据的积累和时间段数的增多,虽然异常实例总数也有所增加,但其相关性建模计算结果较为可靠,因此能对异常部位精准识别.



(a) 准确率对比 (b) 召回率对比

Fig.13 Varying total number of anomaly instances

图 13 异常总量变化对算法性能的影响



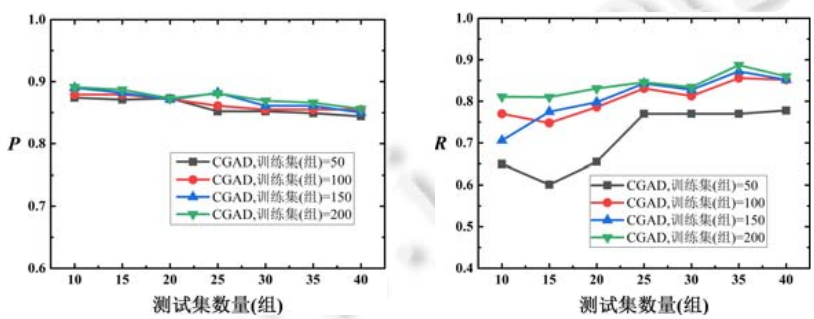
(a) 准确率对比 (b) 召回率对比

Fig.14 Varying size of training set

图 14 测试集规模对算法性能的影响

(4) 训练集规模的影响

我们在图 15 中介绍了训练集规模对本文 CGAD 算法识别效果的影响.



(a) 准确率对比 (b) 召回率对比

Fig.15 Varying size of test set for CGAD

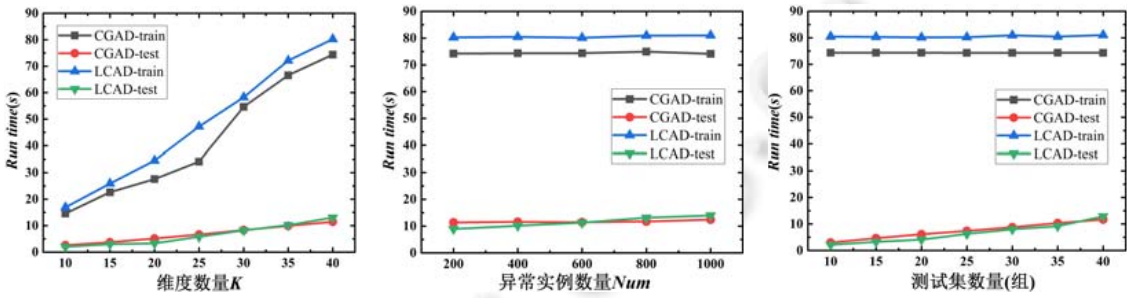
图 15 训练集规模对 CGAD 算法性能的影响

在相同测试集规模情况下,我们分别测试了以 50,100,150,200 个序列时间组为训练集个数的训练过程对计算结果的影响.从图 15(a)可以看出,随着测试集数据的增多,这 4 组实验的 P 值均有小幅度下降;且相同情况下,训练集数量多时,P 值更高.但总体而言,4 组实验的 P 值比较接近.图 15(b)显示,4 组实验的 R 值均有小幅增长且趋于稳定.但训练集样本数量对 R 值影响较大:当训练集样本数量较少时,R 值较低;当训练集的时间组数超过 100 后,R 值较高.实验结果说明,本文提出的 CGAD 算法的识别准确率较为稳定,利用较少训练集样本即可实现

高质量的异常检测任务,当训练集规模足够时,能够尽可能地查全异常问题.

5.3 方法效率分析

本节介绍了 CGAD 算法和 LCAD 算法在训练和检测阶段所需计算用时效率上的对比实验,在图 16 中介绍各个参数对时间效率的影响,在图 17 中介绍了 CGAD 算法的训练集规模对其检测时间的影响.



(a) 序列维数总数 (b) 异常实例总数 (c) 测试集规模

Fig.16 Varying parameters for efficiency evluation

图 16 算法效率对比实验

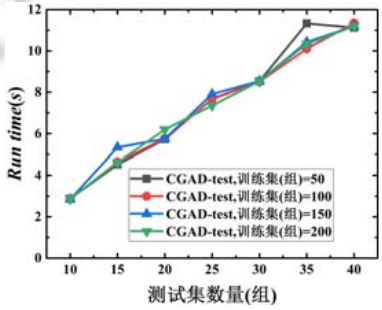


Fig.17 Efficiency evaluation for varying training size

图 17 4 组不同训练集规模对算法效率的影响

图 16(a)~图 16(c)分别介绍了序列维数、异常模式总数量、测试集规模对异常检测任务所用时间的影响.从图中发现,CGAD 算法和 LCAD 算法在训练过程的用时均大于在检测过程的用时.主要原因是训练集样本较大,相关性建模需要花费时间计算.从图 16(a)得知,在相同条件下,CGAD 算法的训练时长略低于 LCAD 算法.CGAD 算法在建立时序相关图模型时需要进行图上的计算,但由于序列维数不大,因此图的规模不会很大,计算量有限.因此在检测过程中,其用时在 40 维数据内几乎呈线性增长.

从图 16(b)、图 16(c)得知,异常实例总数和测试集数量的变化未给两种算法的训练的时间开销带来明显的影响,这表明两种算法在复杂的混有许多异常数据的时间序列上具有可扩展性和可靠性.但这两个参数的增长让测试过程的用时增加,这是由于异常实例的增多,增加了相关性图上的计算次数.总体来看,本文的 CGAD 算法检测用时与 LCAD 算法接近,且随着数据量的增加,LCAD 算法耗时有超过 CGAD 算法的趋势.这说明本文算法在比 LCAD 算法查的更准和更全的同时,通过对样本数据的训练,CGAD 算法的检测用时和时间复杂度已明显低于需遍历所有序列维数,且有高计算复杂性的常规异常检测算法(由于篇幅所限,本文未展示此部分实验结果).因此,本文的 CGAD 算法满足了方法效率和效果的平衡.

图 17 介绍了在 40 维序列中 600 个异常实例的情况下,分别以 50,100,150,200 个序列时间组作为训练数据时,4 组实验对应的检测过程所用时间的变化.可以看出:随着训练集数量的增长,4 组实验的检测用时几乎线性增长;且相同条件下,4 组实验的用时非常相近,几乎重合.这说明训练集的规模对测试过程的时间花销几乎没有影响.结合上述实验分析,这启示了我们:在时间允许的情况下,可以通过增加训练集样本数量来实现更准确可

靠的训练过程,进而在异常检测过程中实现高效率和精准全面的检测.

6 总结和展望

本文研究了基于相关性分析的时序数据异常检测方法,提出了解决该问题的方法框架,并结合案例分析,分别介绍了多维时间序列的相关性计算算法和基于相关性的异常检测算法.通过在真实制造业领域数据集上的大量实验,本文验证了所提出方法在解决时序数据异常检测问题的准确性和效率都优于已有的基于统计和基于学习模型的基本方法.未来的研究方向包括:(1) 对于新的异常实例进行反馈学习的异常检测算法研究;(2) 基于有限标签的弱监督异常检测方法研究.

References:

[1] Zhang J, Qin W, Bao JS, *et al*. Big Data in Manufacturing Industry. Shanghai: Shanghai Science and Technology Press, 2016 (in Chinese).

[2] Industrial Big Data Special Unit of Industrial Internet Alliance. Industrial Big Data Technology and Application Practice. Beijing: Publishing House of Electronics Industry, 2017 (in Chinese).

[3] National Manufacturing Strategy Advisory Committee. “Made in China 2025” Technology Roadmap for Key Areas. 2015 (in Chinese).

[4] Wang JM. White Paper on Big Data Technology and Application in China’s Industry. Beijing: Alliance of Industrial Internet, 2017 (in Chinese).

[5] Wang JM. Summary of industrial big data technology. Big Data Research, 2017,(6):3−14 (in Chinese with English abstract).

[6] Li J, Ni J, Wang AZ. From Big Data to Intelligent Manufacturing. Shanghai: Shanghai Jiaotong University Press, 2017 (in Chinese).

[7] Lee J, Wrote; Qiu BH, Trans. Industrial Big Data: The Revolutionary Transformation and Value Creation in INDUSTRY 4.0 Era.. Beijing: China Machine Press, 2015. (in Chinese).

[8] Chandola V, Banerjee A, Kumar V. Anomaly detection: A survey. ACM Computing Surveys, 2009,41(3):15:1−15:58.

[9] Toledano M, Cohen I, Ben Y, *et al*. Real-time anomaly detection system for time series at scale. In: Proc. of the SIGKDD Workshop. 2017. 56−65.

[10] Gupta M, Gao J, Aggarwal C, Han JW. Outlier detection for temporal data. Morgan & Claypool Publishers, 2014,26(9):2250−2267.

[11] Enders W, Wrote; Du J, Xie ZC, Trans. Applied Econometric Time Series. 2nd ed., Beijing: Higher Education Press, 2006 (in Chinese).

[12] Eamoon JK, Kaushik C, Sharad M, Michael JP. Locally adaptive dimensionality reduction for indexing large time series databases. ACM Trans. on Database Systems. 2002,27(2):188−228.

[13] Ma J, Ma J, Perkins S, *et al*. Time-series novelty detection using one-class support vector machines. In: Proc. of the Int’l Joint Conf. on Neural Networks. IEEE, 2003.

[14] Chandola V, Mithal V, Kumar V. Comparative evaluation of anomaly detection techniques for sequence data. In: Proc. of the 8th IEEE Int’l Conf. on Data Mining. IEEE Computer Society, 2008.

[15] Li X, Han J, Kim S, *et al*. ROAM: Rule- and motif-based anomaly detection in massive moving object data sets. In: Proc. of the 7th SIAM Int’l Conf. on Data Mining. 2007. 273−284.

[16] Golab L, Karloff H, Korn F, *et al*. Sequential dependencies. Proc. of the VLDB Endowment, 2009,2(1):574−585.

[17] Song S, Zhang A, Wang J, *et al*. SCREEN: Stream data cleaning under speed constraints. In: Proc. of the ACM SIGMOD Int’l Conf. on Management of Data. ACM, 2015. 827−841.

[18] Zhang J, Wang H. Detecting outlying subspaces for high-dimensional data: The new task, algorithms, and performance. Knowledge and Information Systems, 2006,10(3):333−355.

[19] Papadimitriou S, Sun J, Faloutsos C. Streaming pattern discovery in multiple time-series. In: Proc. of the 31st Int’l Conf. on Very Large Data Bases. 2005. 697−708.

[20] Ghoting A, Parthasarathy S, Otey ME. Fast mining of distance-based outliers in high-dimensional datasets. Data Mining and Knowledge Discovery, 2008,16(3):349−364.

[21] Fujimaki R, Nakata T, Tsukahara H, *et al*. Mining abnormal patterns from heterogeneous time-series with irrelevant features for fault event detection. Statistical Analysis and Data Mining, 2009,2(1):1−17.

[22] Ding J, Liu Y, Zhang L, *et al*. An anomaly detection approach for multiple monitoring data series based on latent correlation probabilistic model. Applied Intelligence, 2016,44(2):340−361.

[23] Zhang A, Song S, Wang J, *et al*. Time series data cleaning: From anomaly detection to anomaly repairing. Proc. of the VLDB Endowment, 2017,10(10):1046−1057.
[24] Ding XO, Wang HZ, Su JX, *et al*. Cleanits: A data cleaning system for industrial time series. Proc. of the VLDB Endowment, 2019,12(12):1786−1789.
[25] Wang M, Zhang C, Yu J. Native API based windows anomaly intrusion detection method using SVM. In: Proc. of the IEEE Int'l Conf. on Sensor Networks. IEEE, 2006.
[26] Gao B, Ma HY, Yang YH. HMMs (hidden Markov models) based on anomaly intrusion detection method. In: Proc. of the 2002 Int'l Conf. on Machine Learning and Cybernetics. 2002. 381−385.
[27] Qiao Y, Xin XW, Bin Y, *et al*. Anomaly intrusion detection method based on HMM. Electronics Letters, 2002,38(13):663−664.
[28] Zhang X, Fan P, Zhu Z. A new anomaly detection method based on hierarchical HMM. In: Proc. of the 4th Int'l Conf. on Parallel and Distributed Computing, Applications and Technologies. 2003. 249−252.
[29] Zhou ZH. A brief introduction to weakly supervised learning. National Science Review, 2018,(1):48−57.
[30] https://en.wikipedia.org/wiki/Pearson_correlation_coefficient
[31] Wang C, Guo ZH, Wang JM. Industrial big data and its technical challenges. Telecommunications Network Technology, 2017,8(8): 1−4 (in Chinese with English abstract).
[32] Cormen TH, Leiserson CE, Rivest RL, Stein C. Introduction to Algorithms. 3rd ed., MIT Press, 2009.
[33] Romeo R. A short proof of Konig's matching theorem. Journal of Graph Theory, 2000,33(3):138−139.
[34] https://en.wikipedia.org/wiki/Hungarian_algorithm

附中文参考文献:

[1] 张洁,秦威,鲍劲松,等.制造业大数据.上海:上海科学技术出版社,2016.
[2] 工业互联网产业联盟工业大数据特设组.工业大数据技术与应用实践.北京:电子工业出版社,2017.
[3] 国家制造强国建设战略咨询委员会.《中国制造 2025》重点领域技术路线图.2015.
[4] 王建民.中国工业大数据技术与应用白皮书.北京:工业互联网产业联盟,2017.
[5] 王建民.工业大数据技术综述.大数据,2017,(6):3−14.
[6] 李杰,倪军,王安正.从大数据到智能制造.上海:上海交通大学出版社,2017.
[7] 李杰,著;邱伯华,译.工业大数据:工业 4.0 时代的工业转型与价值创造.北京:机械工业出版社,2015.
[11] Enders W,著;杜江,袁景安,译.应用计量经济学:时间序列分析.第 2 版,北京:高等教育出版社,2006.
[31] 王晨,郭朝晖,王建民.工业大数据及其技术挑战.电信网技术,2017,8(8):1−4.

丁小欧(1993－),女,黑龙江哈尔滨人,博士生,CCF 学生会员,主要研究领域为数据质量,数据清洗,时序数据挖掘,异常检测.

于晟健(1997－),男,硕士生,CCF 学生会员,主要研究领域为,数据清洗,时序数据挖掘,异常检测,大数据管理.

王沐贤(1997－),男,主要研究领域为数据清洗,异常检测,时序数据库系统.

王宏志(1978－),男,博士,教授,博士生导师,CCF 杰出会员,主要研究领域为数据库,大数据,数据质量.

高宏(1966－),女,博士,教授,博士生导师,CCF 杰出会员,主要研究领域为复杂结构数据管理,无线传感器网络.

杨东华(1976－),男,博士,副教授,博士生导师,主要研究领域为大数据管理与分析.