MDCPD:基于矩阵序列距离度量的数字生态变点检测

朱业琪 刘明义 苏统华 王忠杰

(哈尔滨工业大学计算学部 哈尔滨 150001)

摘 要 数字生态系统是一个分布式的、适应性的、开放的社会技术系统.随着大数据、物联网、云计算等技术的发展,数字生态的表现形式逐渐复杂多样,与人们的生活更加密切.数字生态受内外部激励自发性地持续演化,一些事件的发生可能会使数字生态的部分性质显著变化,偏离其正常的演化路径,进而导致生态伴随着异常不健康地发展,如果能够及时发现这些变化并定位引起变化的事件,然后加以人为干预,则可能将负面影响降到最低.动态复杂网络是一个辅助观测数字生态的有效工具,这使分析生态的演化情况成为可能,复杂网络分析领域中的变点检测是检测数字生态演化变点的主要技术手段之一.然而,目前已有的通用的变点检测方法未针对数字生态做出优化,忽视了数字生态的高度动态等特性,会导致这些方法可能无法在高度动态、持续变化的情况下检测变点,于是,已有方法在数字生态场景上的变点检测性能可能不佳.为解决上述问题,本文提出基于矩阵序列距离度量的数字生态变点检测方法(MDCPD),MDCPD 是社区感知的,它从数字生态的社区视角观测数字生态的变化幅度,通过计算社区矩阵距离变化率在在连续时间动态网络建模的数字生态上高效地实现了变点检测,且变点检测和数字生态演化动因定位均是事件级别,能帮助生态的管理人员高效地进行干预和决策.为抵抗社区结构矩阵序列数据中的噪声对方法的影响,本文提出了矩阵干预策略,通过从数字生态中观测到的客观条件干预社区结构矩阵的数值,提高了社区结构矩阵序列对数字的生态结构表达能力.本文在基于合成数据的连续时间和离散时间两个场景的对比实验以及消融实验证明了 MDCPD 和矩阵干预策略的有效性,MDCPD 的 F1 指标至多超过 SOTA 方法 0.383,矩阵干预策略至多使 MDCPD 的 F1 指标提高了 0.053.最后,本文在真实数字生态数据集上进行了案例分析,进一步说明了 MDCPD 的实践价值.

关键词 数字生态;变点检测;动态网络;复杂网络分析;异常检测

中图法分类号 TP301 **DOI 号** 10.11897/SP.J.1016.2024.02452

MDCPD: Change Point Detection for Digital Ecosystem Based on Sequenced Matrices Distance Measurement

ZHU Ye-Qi LIU Ming-Yi SU Tong-Hua WANG Zhong-Jie

(*Faculty of Computing, Harbin Institute of Technology, Harbin* 150001)

Abstract A digital ecosystem is a distributed, adaptive, and open social-technical system that possesses characteristics similar to natural ecosystems, such as self-organization, scalability, and sustainability. With the development of technologies such as big data, Internet of Things, and cloud computing, the concept of digital ecosystem has become increasingly complex and diverse, becoming more closely related to people's lives. Digital ecosystem evolves continuously due to the internal and external impacts. Some events may significantly change the properties of the digital ecosystem and make it deviates from its normal evolutionary path, thus causing the ecosystem

收稿日期:2023-11-11;在线发布日期:2024-07-03.本课题得到国家重点研发计划资助项目(2021YFB3300700)、国家自然科学基金资助项目(62372140,62277011)资助.**朱业琪**,博士研究生,中国计算机学会(CCF)学生会员,主要研究方向为服务计算、复杂网络分析、图神经网络、数据挖掘.E-mail:yqzhu@stu.hit.edu.cn.**刘明义**,博士,助理教授,中国计算机学会(CCF)会员,主要研究领域为服务生态、服务演化分析、数据挖掘、图神经网络.**苏统华**(通信作者),博士,教授,中国计算机学会(CCF)会员,主要研究领域为模式识别、手写汉字识别、深度学习、异构计算.E-mail:thsu@hit.edu.cn.**王忠杰**,博士,教授,中国计算机学会(CCF)会员,主要研究领域为服务计算、软件工程.

unhealthily evolve with anomalies. If these changes can be detected and human intervention is carried out in time, the negative impact may be minimized. Digital ecosystem is observable through event flow, which naturally forms a network structure and makes it possible to analyze the evolution status of a digital ecosystem. Under this consideration, it is a mainstream to use the complex network to model a digital ecosystem, and change point detection in such field is one of the main techniques to detect evolution status of a digital ecosystem. However, few existing approach on change point detection are optimized for the characteristics of the digital ecosystems. They often overlook the dynamics property as they prefer discrete-time dynamic graph for modeling. This results in they are not able to support the detection of change points in a digital ecosystem under dynamic and continuous changing scenario, which brings the decrease of the performance in change point detection task for digital ecosystem. To address the problem, in this paper, we propose Change Point Detection for Digital Ecosystem Based on Sequenced Matrices Distance Measurement (MDCPD). MDCPD is a community-aware approach, which utilizes the community structure matrix sequence efficiently to assess the happened changes from community perspective. Therefore, MDCPD can detect change points in digital ecosystem under continuous-time dynamic network modeling as well as locate the reason for evolution at event-level. Based on two distance measurement methods, we develop two variants of MDCPD, and their advantages are illustrated in the paper. To reduce the noise in the matrix sequence, we propose the matrix intervention strategy. The strategy enhances the expressivity of community structure matrix sequence according to the weight matrix observed in the digital ecosystem, which improves the performance of MDCPD. We show the effectiveness of MDCPD through the comparison with the state-of-the-art (SOTA) approaches on both continuous and discrete scenarios, where each scenario contains two synthetic datasets. MDCPD exceeds the SOTA approaches by 0.383 and 0.034 on F1 metric on continuous and discrete scenario respectively. An ablation study is carried out on the same datasets to show the effectiveness of matrix intervention strategy. The intervention strategy improves the performance of MDCPD by 0.053 on F1 metric at most. Finally, a case study is conducted on a real digital ecosystem dataset. We demonstrate how MDCPD can be used to analyze digital ecosystem evolution and the potential conclusions that can be drawn by a combination of text and visualization. This suggests the practical value of MDCPD.

Keywords　digital ecosystem; change point detection; dynamic network; complex network analysis; anomaly detection

1　引　言

随着大数据、物联网、云计算等技术的发展，数字生态的概念与我们的生活联系得愈加紧密. 数字生态系统是一个分布式的、适应性的、开放的社会技术系统，具有与自然生态系统类似的自组织性、可伸缩性和可持续性等特性[1]，从内部的实体来看，它是技术和商业结合的. 数字生态的概念本身虽然抽象，但数字生态的表现形式和应用场景十分具体，例如社交网络、服务生态、智慧园区等[2-5]，学者们从数字生态的性质、演化特点、数据特征等多方面对数字生态实例开展了研究，例如 Xue 等人[3-4]和 Liu 等人[7]分析了服务生态演化路径；Rong 等人[8]分析了物联网如何引导商业生态系统共同发展并给出了构建基于物联网的商业生态系统关键思路；王莉等人[2]分析了社交网络生态的演化，介绍了社区检测、异常群体检测等工作并指出了社交网络生态上的工作难点和未来趋势.

不同于开源生态[6,9]这样的由开源软件、硬件、数据、社区等共同组成的系统，数字生态因其自发性和高度不确定性，容易产生异常事件和难以预测的变动，而开源生态在社区和参与者的强监督下运作，通过严格的代码审核和持续的社区协作维护，确保

了高质量和长期可靠性,相对稳定,异常事件较少.

数字生态受内外部激励自发演化,但演化路径充满了不确定性,生态内部的实体不断新增或消失,相互之间的关系也逐渐变化,人们期望得知这些变动是否遵从着特定领域的客观规律健康发展,于是,对数字生态进行快速准确的健康状态判断成为了一个研究热点.若一个生态在异常因素影响下持续演化,可能引起许多问题,例如,Liu 等人[10]对 Web 服务生态的研究指出,ProgrammableWeb① 平台上的 Web 服务没有被良好地维护,许多服务失效却依然被标记为可用,导致该平台收录的 Web 服务整体质量下降,且组成的 mashup 功能无法达到用户的预期,目前该平台已关停.如果能及时发现这个生态潜在的问题并加以人为干预,则有希望使该生态健康发展.

数字生态是一个复杂的动态网状系统,复杂网络分析是研究网状系统的一个常用技术,已被应用于许多网状系统中,例如社交网络演化分析[11-12],基于分子拓扑结构的化合物功能预测[13-14]、微服务系统异常检测[15].从网络分析的视角观察数字生态,生态中实体的新增或消失、关系的建立或中止等现象在网络中存在与之对应的事件,如新节点的加入、已有节点的消失、节点本身的属性变化,再如边的新增、权重变化、边的消失等.换言之,生态中的部分关键属性和要素的变化能够通过网络结构变化来体现,这表明数字生态演化过程中的许多事件是可观测的,因此,研究网络结构和性质的变化是推测生态的演化路径的可行方式.

变点(Change point)检测是复杂网络分析中围绕动态网络展开的常见问题之一[16].变点体现了网络性质的重大变化,变点前后的网络结构、属性通常会有显著区别,变点检测问题在数字生态上同样成立,通过观测生态的拓扑结构等特征的变化,可以分析数字生态演化的当前状态并判断是否健康发展.社区(Community)作为一种中观(Mesoscopic)视角[2],介于宏观和微观之间,是观测网络的拓扑结构的一个良好媒介,观测数字生态演化时,社区有易于表达和追踪的优点:在用图网络对生态建模后,把每个实体划分到某个社区,该生态如何演化便能够表达,社区结构的变化过程就体现了网络的演化过程,数字生态的变点检测亦是如此,可以通过捕获社区结构的变动,衡量其变化幅度来检测数字生态的演化进程中是否存在变点.

生态中持续发生的事件会影响生态的拓扑结构,进而导致原有的社区结构不再支持描述当前生态并产生演化,当这个时刻的社区结构变动较大时,认为该时刻为变点.图 1 展示了这个过程,$t_1 \sim t_3$ 的网络示意了一个生态的演化进程,属于相同社区的节点被同一条曲线包围,其中 t_3 时刻展示了可能由 t_2 时刻的网络变化后的两种情况,虚线为新增的边.情况 1 的社区结构变动较大,社区结构出现了显著变化,因此认为情况 1 的 t_3 是变点,而情况 2 的社区结构变动较小,认为情况 2 的 t_3 时刻生态仍处于正常演化.

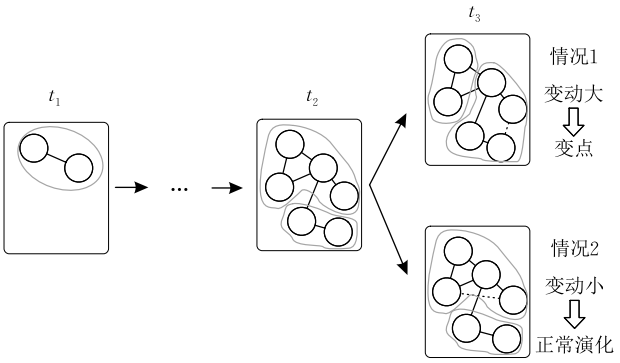


图 1 生态演化过程中变点产生示意

动态图网络的建模有两种:连续时间动态网络(Continuous-Time Dynamic Graphs,CTDG)和离散时间动态网络(Discrete-Time Dynamic Graphs,DTDG)[11],图 2 展示了两种建模方式的区别,"t_i"为连续时间的时间步,"快照i"为离散时间的快照编号.DTDG 通过间隔一段时间取得网络快照来表达动态网络,会忽略掉相邻快照之间发生的事件的先后顺序,而 CTDG 中每个事件都有严格的先后关系.由于数字生态具备高度动态性,数字生态中事件发生十分频繁,所以数字生态对方法的时间连续性有着更高要求,每一个事件都有可能成为生态大幅变动的关键因素,为了连续捕捉数字生态的性质变化,同时在探究演化动因时能够追溯到单个事件,使用 CTDG 进行建模是很有必要的.

然而,目前使用 CTDG 建模数字生态并检测生态变点的研究甚少,而且各方法往往没有针对数字生态的特点如高度动态性做出优化,可能存在准确性差或时间性能不佳的问题.为在连续时间场景上高效解决数字生态的变点检测问题,本文提出基于矩阵序列距离度量的数字生态变点检测方法(Change Point Detection for Digital Ecosystem Based on Sequenced

① https://www.programmableweb.com

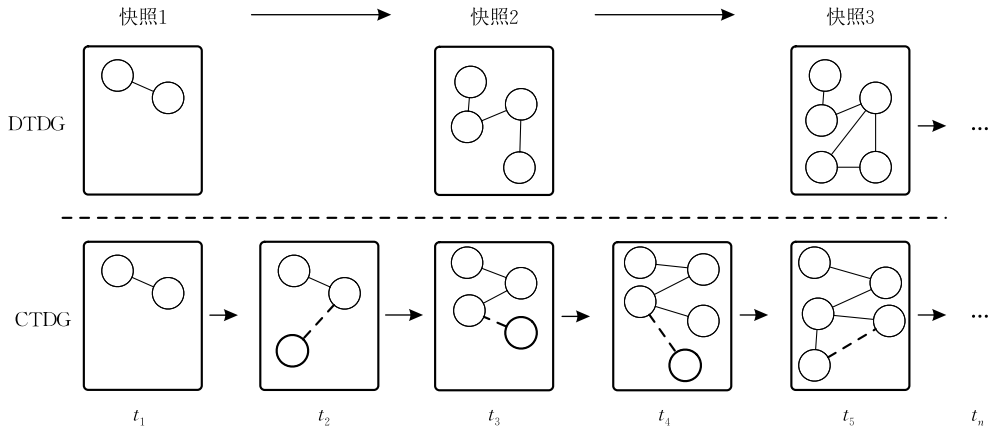


图 2　离散时间建模与连续时间建模的区别

Matrices Distance Measurement，MDCPD），它基于CTDG对数字生态建模，通过衡量社区结构矩阵序列的变化幅度检测变点，以判断每个事件是否引起了生态的大幅变动. 该工作可以揭示复杂网络中的关键转折点和动态变化，为动态网络建模和行为分析提供新的视角，从而帮助人们识别和理解生态中结构性变化和突发事件以做出优化决策，保障生态质量. 具体而言，本文的贡献如下：

(1) 提出了 MDCPD，该方法通过度量矩阵距离高效判断数字生态变点的存在与否，能够同时支持CTDG 和 DTDG 的变点检测，可够根据实际情况调整检测粒度.

(2) 针对数字生态社区结构矩阵序列的噪声问题提出了干预矩阵策略，该策略一定程度上减少了矩阵序列中的极端值，有助于 MDCPD 更好地捕获变点，提升了方法的准确率.

(3) 在两组合成数据上与 SOTA (State-Of-The-Art) 方法进行了对比实验，说明了 MDCPD 的有效性；在合成数据上对矩阵干预策略开展的消融实验证明了这个策略的有效性. 此外，本文还在真实的数字生态数据集上进行了案例分析，进一步说明了MDCPD 的输出结果是可靠的，表明了 MDCPD 的实践价值.

本文第 2 节介绍变点检测的有关工作；第 3 节详细介绍本文提出的方法 MDCPD；第 4 节阐述在合成数据集上针对方法有效性的实验设置和实验结果分析；第 5 节在真实的数字生态数据集上进行案例分析；第 6 节总结本文并分析未来展望.

2　相关工作

变点检测旨在检测一个动态网络在其演化进程中是否发生了显著的结构和性质变动，它的技术路线多样，本节主要按照方法所支持的时间连续性作分类并进行述评，最后简要讨论了图神经网络 (Graph Neural Network，GNN) 在变点检测上的运用.

2.1　离散时间的变点检测

Donnat 等人[11]讨论了通过汉明距离和杰卡德距离量化结构的局部变化，通过谱方法对比图结构的演化；Wang 等人[17]在离散的动态网络建模方式的基础上，提出了基于最大似然估计的方法检测变点；Sulem 等人[18]通过图相似度学习实现了这个目标，且该方法不需要任何先验条件，是完全无监督的；Xu 等人[19]通过研究统计相容性，实现了动态网络的社区估计和变点检测，在多个数据集上取得了不错的检测结果，不过，该方法主要适用于无权无向图.

Miller 等人[20]提出的 SizeCPD 通过度分布来捕获快照的特征，且支持滑动窗口，滑动窗口有助于减少假阳性和假阴性的结果，该方法支持在线场景，但检测粒度受限于 DTDG，只支持离散的变点检测；Zhu 等人[21]提出的 CICPD 也采用了滑动窗口的策略，通过节点重要性获得了节点表示，并基于社区检测来判断变点，比许多现有方法更高效、更准确；Cheng 等人[22]提出的方法支持在大规模网络上检测重叠社区和相应的社区变化.

衡量相邻两个时间步的图网络变化情况来判断演化状况是十分符合直觉的方法，这类方法的优化方向主要有两项：一是提升研究对象对图网络的表达能力，例如改进图网络表示向量的学习方式；二是提升向量变化计算方式，要求这些数值能够充分体现出向量的变化情况. 例如 CDP[23]改进了特征向量的获取方式，通过普氏距离为每次变化计算分数，但是对每个时间步的图网络都进行这个操作对资源消耗较大；CPDCN[24]采用二范数来衡量变化情况.

2.2 连续时间的变点检测

以上列举的方法均使用 DTDG 建模动态网络，对于数字生态系统这样高度动态的系统而言，这样的检测粒度是不够的，而如果人为使 DTDG 的快照划分粒度更小，往往会使方法的时间性能大幅增加，在这种情形下，使用 CTDG 对数字生态建模并连续地检测变点是更好的选择.然而，目前仅有少量的方法支持连续时间的变点检测.

RDPG[25]是一个线下线上结合的方法，需假设能够获取到一个干净无变点的历史数据，先通过线下分析，当新的事件发生后，实时监测网络的变化，这个过程是线上的，但是其线下过程是耗时的.这样的折中使得该模型的综合时间性能比较好；He 等人[26]通过矩阵分解的方式对矩阵序列检测变点，由于图网络中的节点特征、邻接矩阵等均以矩阵形式表达，所以该方法可用于 CTDG 的动态图上，且具有较好的可解释性，TILES[27]是一个基于 CTDG 建模的动态网络变点检测方法，它是完全无监督的，支持输入事件流并基于社区结构检测变点，进一步的，该方法将检测到的变点作了更细致的分类，可用性相对更高.

2.3 GNN 在变点检测上的应用

GNN 是研究图网络特征和性质的深度学习技术，GNN 可以充分利用网络结构、节点、边等多种特征，能够用于许多任务，例如吴越等人[28]在图卷积网络的基础上提出异质超图卷积网络模型，用于节点分类；Hao 等人[29]将图注意力网络运用于社区检测工作；STP-GCN[30]、DAuCNet[31]、FairHELP[32]均采用 GNN 获得了表达能力优秀的向量并用于链接预测的工作.

将 GNN 用于变点检测的主要难点是对时间维度的处理，即如何让 GNN 利用历史信息并让学习到的向量存在时间关系.Ryck 等人[33]通过自编码器和时间不变性，在时间序列的数据上实现了变点检测的工作；Zhu 等人[34]提出了时序图卷积网络并在动态网络上实现了检测变点的方法.

总的来说，变点检测任务面向一个动态过程，所有方法均要对图网络动态建模，建模方式以 CTDG 和 DTDG 为主，通过研究邻接矩阵、社区结构等网络属性和特征，现有方法在部分具有网络结构的数据上取得了较好结果.然而，目前并无针对数字生态提出的演化变点检测方法，数字生态的一些特点如高度动态、语义信息等将不能被很好地被建模进去，从而导致在数字生态上做变点检测时准确性下降.

3 基于矩阵序列距离度量的变点检测方法

为了加强对数字生态的特点的捕获并在数字生态上解决变点检测问题，本文提出基于矩阵序列距离度量的数字生态变点检测方法(MDCPD)，该方法通过衡量社区结构的变化幅度检测数字生态的变点，因它的研究对象是生态在演化过程中连续变化的属性，所以 MDCPD 能够很好地考虑数字生态高度动态的性质并充分运用.

距离是数据挖掘、机器学习和模式识别等领域中常用的概念[35]，用于衡量不同数据点之间的相似度或差异度，进而进行分类、聚类、降维等任务.距离的度量方法丰富，常见的度量如包括马氏距离、欧式距离、曼哈顿距离、切比雪夫距离、余弦相似度等.每种距离度量方法都有其独特的优劣和适用场景.

本节首先给出数字生态及问题定义，然后介绍 MDCPD 的细节，表 1 展示了本文将使用的关键符号.

表 1 符号表

符号	含义
$\mathcal{T}$	数字生态系统目前已被观测到的时间步总数
$\mathcal{G}$、$\mathcal{G}^{(t)}$	动态数字生态系统、时间步 t 的数字生态快照
$DE^{(\mathcal{T})}$	总共有$\mathcal{T}$个时间步的，按照时间排序的事件序列
$e^{(t)}$	t 时间步发生的事件
$\gamma(\cdot)$	变点检测任务
$\boldsymbol{D}$、$\boldsymbol{D}^{(t)}$	权重矩阵、时间步 t 的权重矩阵
$\boldsymbol{F}$、$\boldsymbol{F}^{(t)}$	社区结构矩阵、t 时间步的社区结构矩阵
$d^{(t)}$	时间步 t 与 $t-1$ 的矩阵距离
dis	社区矩阵距离序列
$distance(\cdot)$	距离函数
cr	矩阵变化率序列
$cr^{(t)}$	时间步 t 与 $t-1$ 的矩阵距离变化率

3.1 问题定义

本小节给出本文的研究对象：CTDG 建模下的数字生态系统、变点、变点检测.

定义 1. (动态)数字生态系统 $\mathcal{G}=DE^{(\mathcal{T})}$，其中 $DE^{(\mathcal{T})}=\{e^{(0)},e^{(1)},\cdots,e^{(t)},\cdots,e^{(\mathcal{T})}\}$ 是一个按照时间排序的事件序列，t 称作时间步，一个时间步上仅发生一个事件；$\mathcal{T}$ 表示该动态数字生态系统目前已被观测到的时间步总数.事件 $e^{(t)}=(u^{(t)},v^{(t)},w^{(t)})$，当一个事件发生时，等效于网络中新增了一条有向边，其源节点和目标节点分别是 $u^{(t)}$ 和 $v^{(t)}$，$w^{(t)}$ 是边的权重.固定时间步为 t 时，取该网络的切片 $\mathcal{G}^{(t)}=(V^{(t)},E^{(t)})=DE^{(t)}$ 即是静态网络，其中 $V^{(t)}$ 为节点集合，$E^{(t)}$ 为边集合.

定义 2.　变点和变点检测. 给定一个动态数字生态系统$\mathcal{G}$，取连续两个时间步的数字生态系统快照$\mathcal{G}^{(t-1)}$和$\mathcal{G}^{(t)}$，$\mathcal{G}^{(t-1)}$和$\mathcal{G}^{(t)}$的社区划分分别是$c^{(t-1)}$与$c^{(t)}$，若$c^{(t-1)}$与$c^{(t)}$的结构不同(如社区出现了合并、分裂、消失、节点变动等)，则认为t时间步为变点. 变点检测任务意为判断每个时间步是否是变点，可以形式化表示为$\gamma(t)\rightarrow\{0,1\}$，$t\in\{0,1,\cdots,\mathcal{T}\}$，其中 0 表示该时间步不是变点，1 表示该时间步是变点.

3.2　方法细节

动态标签传播算法[36](Dynamic Label Propagation Algorithm，DyLPA)从集团企业数字生态的社区检测出发，实现了 CTDG 建模上的连续时间社区检测，具体而言，DyLPA 在所有时间步都能够输出一个社区结构矩阵，该矩阵通过节点的社区分配情况表达了节点特征以及社区结构，任意两个相邻的时间步上输出的矩阵具有时间连续性. 连续取得多个时间步上社区结构矩阵并组织为矩阵序列，即可表达社区结构的演化路径和演化特征.

DyLPA 的执行过程分别局部传播和全局传播，其中，局部传播指在每个事件发生时，局部更新权重矩阵、概率转移矩阵和社区结构矩阵，全局传播则进行高阶的信息交换. 由于局部传播的输出更加突显数字生态的持续变动过程，而全局传播或破坏社区结构矩阵的连续性质，故本文不执行全局传播过程.

以第t个时间步为例，局部传播的流程可表示为

$$\boldsymbol{D}_{u,v}^{(t)}=\boldsymbol{D}_{u,v}^{(t-1)}+w^{(t)} \tag{1}$$

$$\boldsymbol{P}_{u}^{(t)}=\boldsymbol{D}_{u}^{(t)}\Big/\sum\boldsymbol{D}_{u}^{(t)} \tag{2}$$

然后，重复数次社区结构矩阵的局部更新操作或直到收敛：

$$\boldsymbol{F}_{u}^{(t)}=\boldsymbol{P}_{u}^{(t)}\cdot\boldsymbol{F}^{(t-1)} \tag{3}$$

$\boldsymbol{F}^{(t)}$是社区结构矩阵，将所有已观测到的$\boldsymbol{F}^{(i)}$，$1\leqslant i\leqslant\mathcal{T}$组成序列$\boldsymbol{F}$即是本文的研究对象. 图 3 示意了 MDCPD 的完整过程，算法 1 为 MDCPD 的伪代码. 通过距离函数$distance(\cdot)$可衡量$\boldsymbol{F}^{(t)}$和$\boldsymbol{F}^{(t-1)}$的距离，该距离以一个数$d^{(t)}$来表达，该值越大，认为在t时刻，社区有越大的概率发生变动，进而认为是变点的概率就越大. 将矩阵序列的任何两个相邻的矩阵都计算距离，则能够获得一个矩阵距离序列$dis=\{d^{(1)},d^{(2)},\cdots,d^{(\mathcal{T})}\}$，其中，令$d^{(1)}=0$.

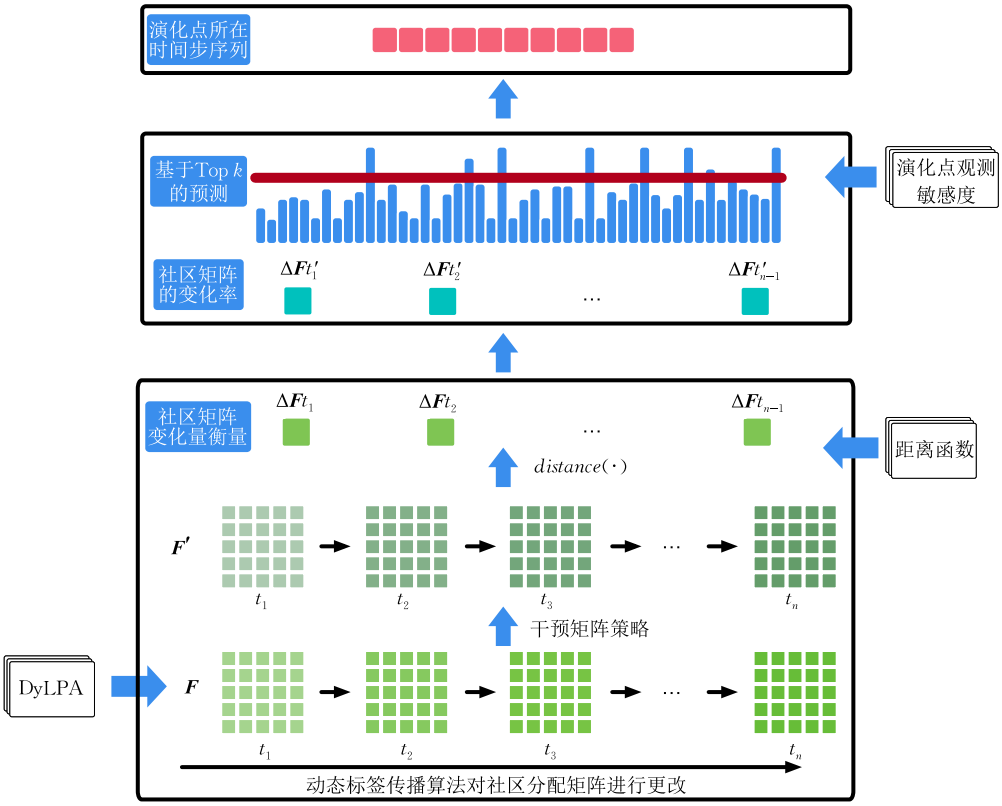


图 3　基于矩阵序列距离度量的数字生态变点检测方法框架

算法 1.　MDCPD.

输入：社区结构矩阵序列$\boldsymbol{F}$；权重矩阵$\boldsymbol{D}$；变点数目k

输出：变点所在时间步编号

1. 初始化$\boldsymbol{F}_s$；
2. FOR $i=0$ to $\boldsymbol{F}.length-1$ DO
3. 　$\boldsymbol{F}_s[i]\leftarrow INTERFERE(\boldsymbol{F}[i],\boldsymbol{D}[i])$；

4. END FOR
5. $\boldsymbol{F}_s.insert(0,\boldsymbol{0})$;
6. $\boldsymbol{F}_s.delete(\boldsymbol{F}_s.length-1)$;
7. 初始化 dis;
8. FOR $i=1$ to $\boldsymbol{F}_s.length-1$ DO
9. $dis[i-1]=distance(\boldsymbol{F}_s[i],\boldsymbol{F}_s[i-1])$;
10. END FOR
11. 初始化 cr;
12. FOR $i=2$ to $\boldsymbol{F}.length-1$ DO
13. $cr[i]=(dis[i+1]-dis[i])/dis[i]$;
14. END FOR
15. RETURN $\arg\max(cr,k)$;

但是,变点的产生不是一蹴而就的,它是由一个或多个事件累积形成的结果,$\boldsymbol{F}$ 在变点之前会持续不稳定,所以矩阵距离会持续较大,因此,采用变化率 $cr^{(t)}=\dfrac{d^{(t+1)}-d^{(t)}}{d^{(t)}}$,$t\geq 2$ 来描述 t 时间步是变点的可能性,于是,矩阵的变化率序列可以表示为 $cr=\{cr^{(1)},cr^{(2)},\cdots,cr^{(\mathcal{T}-1)}\}$,其中,令 $cr^{(1)}=0$. 最后,为了控制算法的敏感度并做出合适规模的预测,需要给定变点数目 k,意为该数据中存在 k 个变点,于是根据 cr 选出最大的 k 个值所对应的时间步作为最终输出结果.

3.3 距离函数的选取

式(4)为矩阵 $\boldsymbol{A}\in R^{N\times N}$ 和 $\boldsymbol{B}\in R^{N\times N}$ 之间的闵可夫斯基距离:

$$distance(\boldsymbol{A},\boldsymbol{B})=\sum_{i=0}^{N-1}\sum_{j=0}^{N-1}(|a_{ij}-b_{ij}|^{p})^{\frac{1}{p}} \tag{4}$$

其中 a_{ij} 和 b_{ij} 分别是矩阵 $\boldsymbol{A}$ 和 $\boldsymbol{B}$ 的第 i 行第 j 列的元素. 当 p 取不同值时,闵可夫斯基距离能衍生出不同的距离函数,它们也代表不同的物理意义,能表达出不同的距离衡量方法,例如,$p=1$ 时,该距离称为曼哈顿距离,表达了一个空间中两个点从平行于坐标轴的方向上的距离之和;$p=2$ 时称为欧式距离,表达了两个点在空间中的直线距离. 图 4 展示了 p 在不同取值时,基于不同的距离度量方式,以原点作为圆心的单位圆的变化.

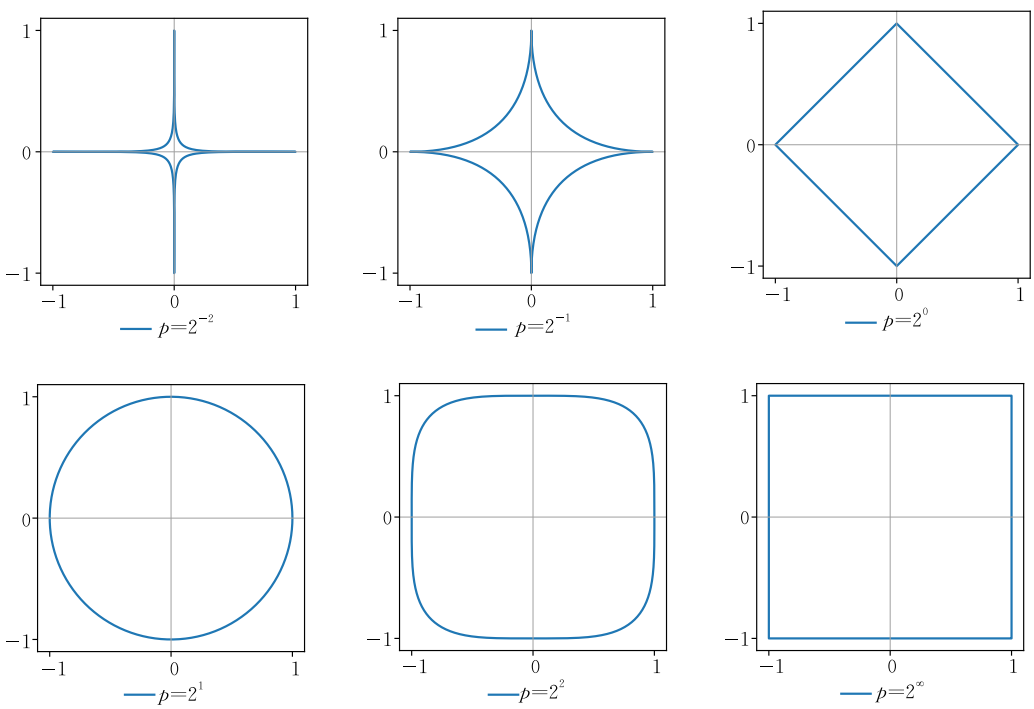


图 4 不同的距离度量方式的单位圆的变化

在实际应用中,为了更好地突显矩阵之间的差异,可以根据矩阵 $\boldsymbol{A}$ 和 $\boldsymbol{B}$ 的特点或条件选择合适的距离函数,本文将 p 取 1 和∞,分别为曼哈顿距离和切比雪夫距离. 具体而言,矩阵的曼哈顿距离是各对应位置元素之差的绝对值之和,可以充分考虑每个元素的差距;切比雪夫距离只考虑各个坐标轴上距离差的最大值,因此更加敏感,但也可能易受极端值的影响.

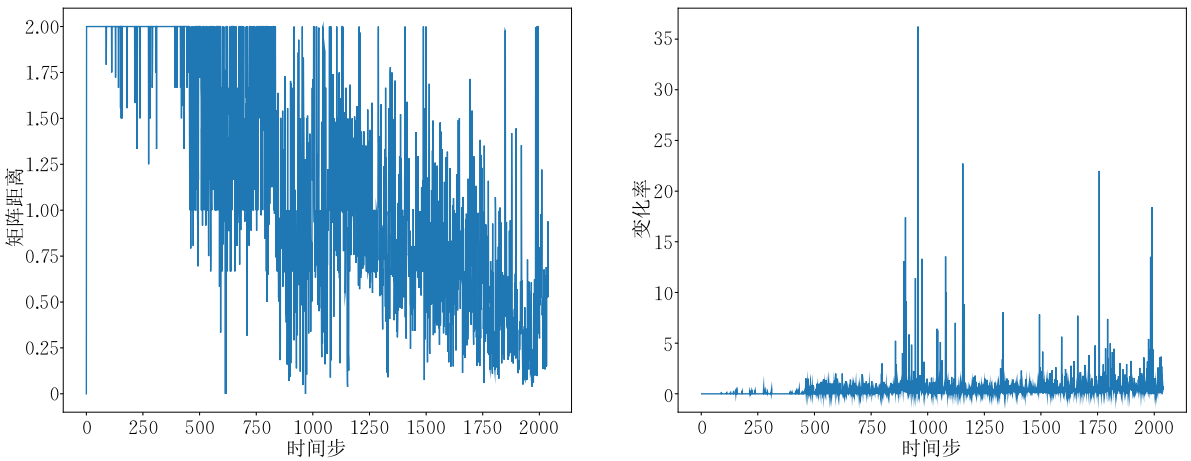
$\boldsymbol{F}^{(t+1)}$ 和 $\boldsymbol{F}^{(t)}$ 是同一场景下不同时刻的社区结构矩阵. 社区结构矩阵从其构造上看虽然是每个节点的向量拼接,但是从矩阵的列的视角来看,它是社区内容向量的拼接,该向量的每个维度相对独立,且每个维度对矩阵的贡献都十分重要. 曼哈顿距离考虑的是各个维度独立的贡献,如城市街区中沿着街

道行走的距离，因此它能够有效地度量这些特征间的差异以及其对整体结果的影响；切比雪夫距离突出的是最大单个维度的距离，如国际象棋中王的移动方式，数字生态中，(度)大节点广泛存在且地位十分重要，社区划分有相当一部分是围绕大节点生成的，大节点的变动信息能影响其周围甚至整个生态，在标签传播的支撑下，我们可以通过局部窥见整体，最大单个维度的距离在这种场景下往往能成为变点产生的决定因素，因而使用切比雪夫距离可以更好地捕捉到这些关键信息.

图 5(a)以切比雪夫距离为例，展示了在一个动态网络演化过程中，矩阵距离和它们之间的变化率随时间步的变化趋势图，两张子图的横轴都是时间步，纵轴分别是矩阵距离和矩阵距离变化率. 该图印证了前文的分析，仅仅通过研究矩阵距离找到变点是比较困难的，因为当社区结构变动的时候，矩阵距离会存在持续较大的情况，但是矩阵距离变化率却能很好地满足观测变点存在与否的需求，由图 5(b)可见，矩阵距离变化率存在部分波峰，这些波峰对应的时间步正是变点，此外还有很多小的波峰，它们对应的时间步上可能存在社区结构变动不明显的小型的演化异常，但不足以能够引起社区结构的大幅变化，即不可被视为变点. 本文认为，这些小型的波峰往往是噪声，因此，如何处理原始数据能够将高的波峰更高，小的波峰更小或者更为平缓成为了需要解决的问题. 为了满足上述目标，下一小节将分析小波峰的来源，并介绍矩阵干预策略，以此消除掉部分小波峰，让大波峰更加明显.

3.4　干预矩阵策略

随着 DyLPA 的运行和输出，不稳定的因素会随之而来，若部分节点，尤其是度大节点在 t 时间步的社区分配情况被预测错误，则它在 $t+1$ 时间步上则可能影响更多节点，造成局部社区结构的错误传播，随着算法的进展，错误将会累积，这会给方法带来一定干扰. 因此，如果直接使用 $\boldsymbol{F}$ 计算 dis、cr 并做变点检测，方法的准确性将受到影响，例如，图 5 中的小的波峰虽然可以认为是生态中发生的变动，但因变动比较微小，其所在时间步上发生的事件不是引起“质变”的关键因素，此时不应认为该时间步是变点.



(a) 矩阵距离随时间步的变化　　(b) 矩阵距离变化率随时间步的变化

图 5　矩阵距离和变化率随时间步变化示意

综上所述，错误的社区结构会给 MDCPD 带来负面影响，为解决此问题，本文设计了干预矩阵策略，如算法 2 所示. 该算法使用矩阵 $\boldsymbol{D}$ 来限制 $\boldsymbol{F}$ 的部分元素值，$\boldsymbol{D}$ 作为权重矩阵，能够客观反映当前网络上的边交互关系.

算法 2.　干预矩阵策略 *INTERFERE*.

输入：社区分配矩阵 $\boldsymbol{F}$；权重矩阵 $\boldsymbol{D}$

输出：矩阵 $\boldsymbol{F}$ 的调优版 $\boldsymbol{F}'$

1. $\boldsymbol{F}' \leftarrow copy(\boldsymbol{F})$;
2. /* 获得每个节点与哪个节点的度最大 */
3. FOR $i=0$ to $\boldsymbol{D}[0].length-1$ DO
4. $\quad j \leftarrow \arg\max(\boldsymbol{D}[i])$;
5. $\quad$ /* 使节点在特定社区重叠 */
6. $\quad c \leftarrow \arg\max(\boldsymbol{F}_j)$;
7. $\quad \boldsymbol{F}'_{ic} \leftarrow \max(\boldsymbol{F}_i)$;
8. END FOR
9. RETURN $\boldsymbol{F}'$;

具体而言，该策略的基本思路是：若与节点 u 连接最紧密(度最大)的节点是 v，那么节点 u 和 v 的社区倾向于是同一个，于是，适当调大节点 u 被分

配到某个社区的概率，而这个概率值取决于它当前已经被分配给的社区的概率，即主动将这两个社区在该节点处重叠而不会破坏它的原本所属，该过程如算法 2 的第 4～7 行所示.

矩阵干预策略通过客观的权重矩阵作为依据，平衡了各个节点的社区分布，减少了极端的社区分布情况，起到了消除了噪声的效果，有助于 MDCPD 算法的进行.

3.5 复杂度分析

本小节将从空间复杂度和时间复杂度两个角度分析 MDCPD 本身的复杂度，此处假设已观测到的时间步总数为$\mathcal{T}$，而每个时间步产生一个矩阵，故矩阵序列的长度为$\mathcal{T}$，矩阵序列中每个矩阵大小设为$N\times N$.

空间复杂度分析. MDCPD 的执行过程分为矩阵干预策略的执行和距离计算，其中矩阵干预策略需要对矩阵序列$\boldsymbol{F}$和$\boldsymbol{D}$进行处理，总共需要$O(\mathcal{T}N^2)$的空间来存放矩阵，但是 MDCPD 的两个过程均只需要两个相邻的矩阵即可运行，故只需要$O(N^2)$的空间代价，另外，矩阵干预策略可以并行执行，若以M线程执行，则需要$O(MN^2)$的空间代价，综上，MDCPD 的空间复杂度为$O(MN^2)$，其中$M\ll\mathcal{T}$.

时间复杂度分析. MDCPD 需要付出$O(\mathcal{T}N)$的时间代价对矩阵序列中每个矩阵进行距离度量（假设距离的计算是$O(1)$的）. 矩阵干预策略需要对矩阵的每一行进行处理，考虑单线程运行时，其时间复杂度为$O(\mathcal{T}N^2)$，但该策略支持并行执行，以M个线程运行时，其时间复杂度为$O\left(\frac{\mathcal{T}N^2}{M}\right)$，综合这两步，则 MDCPD 的时间复杂度为$O(\mathcal{T}N^2)$，其中$M\ll\mathcal{T}$且$M\ll N$. 此外，MDCPD 上游任务的算法 DyLPA 在每个事件发生时，需要进行时间代价为$O(N^2)$的局部传播，MDCPD 与 DyLPA 是相对独立的计算过程，可以并行计算，二者在每个时间步上执行的计算的时间复杂度均为$O(N^2)$，因此，对于一个拥有$\mathcal{T}$个事件的动态网络而言，MDCPD 的综合时间复杂度为$O(\mathcal{T}N^2)$.

4 实 验

4.1 实验设置

因为目前缺少对每个时间步都标注了变点的数据集，所以，本文使用 RDyn[37] 生成了两组动态网络数据集对 MDCPD 进行有效性验证. RDyn 是一个动态社区和图网络基准库，主要功能是生成复杂动态图网络，它生成的动态网络中每一条边都有标注发生顺序，在特定的时间步上有标注变点. 本文通过 RDyn 生成了两组数据集，数据集的基本统计信息如表 2 所示.

表 2 合成数据集基本信息

属性名称	数据集 D1	数据集 D2
节点数量	500	500
事件(时间步)数量	2041	2783
边数量	1095	903
快照数量	100	100
变点数量	24	17

对比方法包括两个支持离散场景（LAD[38] 和 CICPD[21]）的和一个支持连续场景（TILES[27]）的变点检测方法，LAD 是基于拉普拉斯谱的变点检测方法，具体是对构造出的拉普拉斯矩阵进行奇异值分解，然后通过奇异值做图嵌入并获得了低维图向量；CICPD 通过节点重要性将高维图网络嵌入到低维表示，基于检测社区结构变化判断各个快照是否为变点；TILES 作为一个动态社区检测算法，支持在 CTDG 的建模方式，且能通过检测社区的生命周期来检测动态网络的变点.

为了充分且全面比较各方法的使用场景和性能，实验的性能对比分为两个部分：(1) 在连续时间场景展示 MDCPD 的性能并与 TILES 进行对比，并在连续时间场景上分析 MDCPD 的特点；(2) 在离散时间场景上将 MDCPD 与 LAD 和 CICPD 对比.

此外，本文在连续时间和离散时间两个场景对 MDCPD 进行有关矩阵干预策略的消融实验，以此探究矩阵干预策略的效果. 实验环境为 Python 3.8，所有方法都独立运行五次，取均值作为最后的性能，由于所有方法的实验结果均是稳定的，故实验结果中未标注标准差. 实现代码及数据集均在 GitHub 上开源①. 最后，第 5 节在真实的数字生态上以案例分析的方式展示 MDCPD 的变点检测结果并进行分析.

4.2 评价指标

本实验的方法性能采用$F1$和时间步差值和Δt_{sum}进行评价，其中 $F1$ 为支持差额评价的版本，即设$\mathcal{X}\subset\{1,2,\cdots,\mathcal{T}\}$为算法预测的变点（或离散场景下的快照）序列，$\mathcal{S}\subset\{1,2,\cdots,\mathcal{T}\}$为真实变点，则正阳性样本($TP$)定义为：对所有$\tau\in\mathcal{S}$，$\exists x\in\mathcal{X}$，使得$|\tau-x|\leq$

① https://github.com/NormanZyq/MDCPD

M，其中，$M\geq 0$ 是可接受的误差时间步范围. 在此基础之上，能够类似获得 TN、FN、FP，于是 P、R、$F1$ 的计算遵循式(5). 这样的评价指标表达为“$F1@M$”的格式，如 $F1@5$，该指标的值越高意味着方法的变点检测结果越接近真实情况.

$$P=\frac{TP}{TP+FP},$$

$$R=\frac{TP}{TP+FN},$$

$$F1=2\times\frac{P\times R}{P+R} \tag{5}$$

在离散时间场景中，M 分别设置为 5、3、2、1 并进行评价，因在这些选定的数据集中，快照数量为 100，因此即使将 M 最大取至 5，该允许的误差也不超过 5%，足以客观评价方法的性能；在连续时间场景中，数据集的事件数量均大于 2000，同样以大约 5%为能够接受的最大误差量，于是，在对连续时间场景下的方法性能评估时，M 取 100、50、20 和 5. 以上对 M 的不同选取能够评价方法不同粒度的检测性能.

式(6)定义了 Δt_{sum}，其中 $\boldsymbol{G}$ 和 $\boldsymbol{P}$ 分别是真实结果和预测结果. Δt_{sum} 表达了预测结果与真实结果的最小误差累积，值越小意味着检测结果越精确，这两个集合需要同时是快照序列或者时间步序列.

$$\Delta t_{sum}=\sum_{i=1}^{n}\min_{1\leq j\leq n}|\boldsymbol{G}_i-\boldsymbol{P}_j| \tag{6}$$

4.3　实验结果与分析

表 3 和表 4 分别展示了在 D1 和 D2 数据集的连续时间场景上 MDCPD 和 TILES 的变点检测性能评价结果；表 5 和表 6 分别展示了在 D1 和 D2 数据集的离散时间场景上 MDCPD、LAD、CICPD 方法的变点检测性能评价结果(最优的指标使用粗体标出，下同).

表 3　D1 连续时间场景实验结果

方法	$F1@100$	$F1@50$	$F1@20$	$F1@5$	Δt_{sum}
MDCPD (Manhattan)	**0.840**	**0.659**	**0.480**	0.240	714
MDCPD (Chebyshev)	0.760	**0.659**	0.440	**0.320**	**492**
TILES	0.457	0.457	0.343	0.171	1524

表 4　D2 连续时间场景实验结果

方法	$F1@100$	$F1@50$	$F1@20$	$F1@5$	Δt_{sum}
MDCPD (Manhattan)	0.500	**0.500**	**0.278**	**0.222**	**941**
MDCPD (Chebyshev)	**0.611**	0.444	0.222	0.167	1478
TILES	0.348	0.174	0.087	0.087	4950

表 5　D1 离散场景实验结果

方法	$F1@5$	$F1@3$	$F1@2$	$F1@1$	Δt_{sum}
MDCPD (Manhattan)	**0.720**	**0.680**	**0.600**	0.520	45
MDCPD (Chebyshev)	0.680	**0.680**	**0.600**	0.560	43
LAD	0.348	0.348	0.304	0.304	264
CICPD	0.686	0.600	**0.600**	**0.571**	**21**

表 6　D2 离散场景实验结果

方法	$F1@5$	$F1@3$	$F1@2$	$F1@1$	Δt_{sum}
MDCPD (Manhattan)	0.500	**0.444**	**0.444**	0.333	**56**
MDCPD (Chebyshev)	**0.545**	0.424	0.424	0.303	84
LAD	0.308	0.256	0.256	0.205	190
CICPD	0.435	0.435	0.435	**0.435**	137

基于表 3 和表 4 可知，在连续时间场景的实验中，MDCPD 在各项指标的性能均优于 TILES，具体而言，在 D1 上，MDCPD 的 $F1@100$、$F1@50$、$F1@20$ 指标比 TILES 分别最多高了 0.383、0.202 和 0.137；MDCPD 的 $F1@5$ 指标比 TILES 高了 0.149，Δt_{sum} 低了 1032；D2 上 MDCPD 的各项指标也显著优于 TILES，这表明 MDCPD 不论是粗精度变点检测还是细精度的变点检测能力均强于 TILES.

MDCPD(Manhattan)与 MDCPD(Chebyshev)互有优劣，两种方法采用的距离函数会影响它们的性质，前者能综合考虑演化过程中生态的整体变化，对于粗粒度的变点检测有着较好的准确性，适合当生态有一定的异常累积并最终影响了生态的整体结构的情形；MDCPD(Chebyshev)对变化更加敏感，这使得它容易捕捉细微的变化，且反应速度相对较快，也使得它会错误地估计出额外的异常事件. 综合来看，MDCPD(Manhattan)更适合事件频繁但整体演化较缓的生态，MDCPD(Chebyshev)更适合演化较快的生态，虽然二者的总体性能接近，但若已知一个数字生态的演化性质来选择更合适的方法，将能够获得更优的变点检测结果.

在离散时间场景的实验中，MDCPD 在 $F1@5$、$F1@3$、$F1@2$ 上取得了最优结果，结果表明当允许适当误差时($M\geq 2$)，该方法的性能较好. 此外，MDCPD 在所有指标上均远优于 LAD，主要原因是经 DyLPA 局部传播得到的矩阵 $\boldsymbol{F}$ 对网络特征的描述在变点检测工作上强于 LAD 的研究对象拉普拉斯矩阵.

CICPD 对局部结构的学习相对更加全面，因而它离散时间场景上做精确预测变点检测时性能取得了最优，该方法在 D1 和 D2 数据集上的 $F1@1$ 比 MDCPD 分别高了 0.051 和 0.102. 但 CICPD 的缺陷也十分显著，例如，CICPD 在 D1 数据集上输出了 44

个变点(总共 100 个快照,实际只有 24 个变点),这个误差是难以接受的.

4.4 关于矩阵干预策略的消融实验

矩阵干预策略是为应对数字生态这样高度动态的场景上社区结构的错误累积现象而设计的,本小节开展了关于矩阵干预策略的消融实验,探讨矩阵干预策略在 MDCPD 中的影响,分析该策略在 MDCPD 中对变点检测起到了怎样的贡献.

表 7 展示了在 MDCPD 在两个数据集的两种场景下启用和关闭矩阵干预策略后方法性能的对比,其中"-R"后缀是指方法使用了未经矩阵干预策略修正的矩阵序列;表头中 $F1@M$ ($F1@N$)意为该列若是连续时间场景则对应的评价指标为 $F1@M$,若是离散时间场景则对应的评价指标为 $F1@N$.

表 7 消融实验结果

数据集	方法	$F1@100$ ($F1@5$)	$F1@50$ ($F1@3$)	$F1@20$ ($F1@2$)	$F1@5$ ($F1@1$)
D1-连续	MDCPD(Manhattan)	**0.840**	**0.659**	**0.480**	0.240
	MDCPD-R(Manhattan)	0.800	0.619	0.440	0.240
	MDCPD(Chebyshev)	**0.760**	**0.659**	0.440	0.320
	MDCPD-R(Chebyshev)	0.720	0.619	0.440	0.320
D1-离散	MDCPD(Manhattan)	**0.720**	**0.680**	**0.600**	0.520
	MDCPD-R(Manhattan)	0.667	0.667	0.564	**0.564**
	MDCPD(Chebyshev)	**0.680**	**0.680**	**0.600**	**0.560**
	MDCPD-R(Chebyshev)	0.634	0.634	0.537	0.439
D2-连续	MDCPD(Manhattan)	0.500	**0.500**	**0.278**	**0.222**
	MDCPD-R(Manhattan)	0.500	0.444	0.222	0.167
	MDCPD(Chebyshev)	0.611	0.444	0.222	0.167
	MDCPD-R(Chebyshev)	0.611	0.444	0.222	0.167
D2-离散	MDCPD(Manhattan)	**0.500**	**0.444**	**0.444**	**0.333**
	MDCPD-R(Manhattan)	0.485	0.364	0.364	0.242
	MDCPD(Chebyshev)	**0.545**	**0.424**	**0.424**	**0.303**
	MDCPD-R(Chebyshev)	0.529	0.412	0.412	0.294

相比不使用矩阵干预策略的 MDCPD-R,MDCPD 在多数情况下的性能都有显著提升(或持平),如 D1 数据集的连续时间场景上,MDCPD 的 $F1@100$ 指标至多提高了 0.040,离散场景上 MDCPD 的 $F1@5$ 指标至多提高了 0.053,纵观所有结果,MDCPD(Manhattan)在 D2 数据集的离散场景的 $F1@1$ 指标提升最大,为 0.091.唯一例外的结果为 D1 数据集离散场景上 MDCPD(Manhattan)的 $F1@1$ 指标比 MDCPD-R(Manhattan)低了 0.044.

综上,矩阵干预策略用于 MDCPD 是有效的.经过矩阵干预策略优化的社区结构矩阵序列用于变点检测时,不论在连续时间场景还是离散时间场景,MDCPD 在大多数情况下均有性能提升,尤其是在检测精度较宽松的情况下,变点检测的性能提升得更为显著.

5 案例分析

由于缺乏完全满足实验要求的真实数字生态数据集,前文仅通过合成的方式尽可能地模拟了具备一定数字生态特点的数据集并展开了实验,并未在真实的数据集上观测各个方法的表现情况以及进行客观评价,但为了尽可能展现 MDCPD 对数字生态变点的检测能力和实用价值,本文采用数据集 LSED[①],通过案例分析的方式说明 MDCPD 以及变点检测对现实中的数字生态有何作用.LSED 是通过新闻事件获得的动态服务交互网络构筑的数字生态系统,覆盖了由 2015 年 1 月至 2017 年 12 月的互联网服务交互,共计包括 3818 个节点、7373 条边和 10432 个时间步.

表 8 的"时间步"一列为 MDCPD 的部分输出结果,结合在该时间步上对社区数量和变动情况,可以得到"异常提示"一列,"备注"展示了该异常的具体现象和原因分析.

表 8 案例分析

时间步	异常提示	备注
8	社区诞生 &. 社区变动	海问联合与深圳市两个实体组成新社区.
188	节点社区变动	
190	节点社区变动	
823	社区诞生/分裂	社区数量从 113 变为 114.
1759	社区合并/节点社区变动	所属 2 个社区的近 40 个不同的节点移动到相同社区.

① https://github.com/HIT-ICES/LSED

（续　表）

时间步	异常提示	备注
2375	社区合并/节点社区变动	社区编号 705 大量节点移动至编号 856 社区（如 AI 应用、字节跳动、地震预警）
2409	节点社区变动	开发者、线下商店多个属于不同社区的节点移动到同一个社区.
2905	社区合并/节点社区变动	陌陌、短视频等属于同一个社区的节点分散到不同社区.
3054	社区合并/节点社区变动	云闪付、中国银联等属于同一个社区的节点一起移动到另一个.
5293	节点社区变动	
6183	节点社区变动	
7468	节点社区变动	

如时间步 3054，云闪付、中国银联离开原来所属的社区而移动到另一个，说明第 3054 个事件是导致该情况发生的最关键事件，进而可以分析该事件的发生是否顺应了国家的政策路线，是否合乎地方的经济政策等等. 再如时间步 2375，图 6 展示了该时间步上的事件发生前后，数字生态的变化情况，包括字节跳动在内的企业、服务提供商、服务一同移动到了新的社区，暗示着随着诸如收购、倒闭、发布功能等事件的累积，有一批企业和服务有着与原先不一致的表现，由此现象可以进一步观察并分析是否存在例如垄断等不好的商业行为.

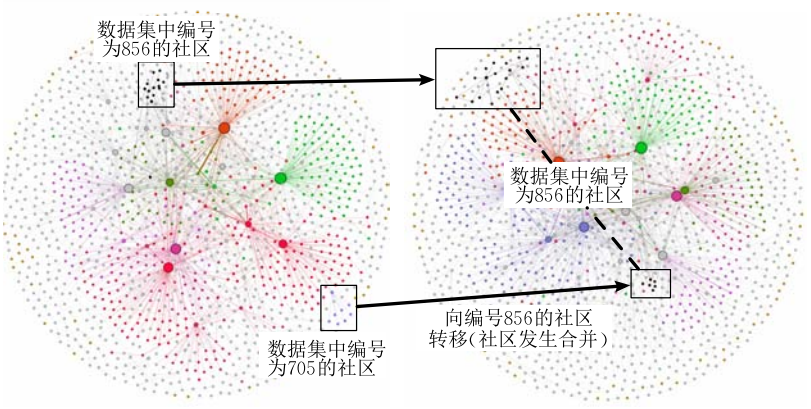


图 6　变点的可视化案例

6　结论与未来展望

本文提出了基于矩阵序列距离度量的变点检测方法 MDCPD，它支持 CTDG 和 DTDG 两种建模方式的动态网络，支持极细粒度和粗粒度的变点检测. 根据使用的距离度量方法不同，衍生了两个变体，MDCPD（Manhattan）和 MDCPD（Chebyshev），根据距离函数性质不同，二者分别有适当的应用场景. 此外，本文提出了矩阵干预策略，用于优化社区结构矩阵序列，减少错误累积带来的噪声. 通过在 RDyn 合成数据集上与 SOTA 方法的对比实验和消融实验，说明了 MDCPD 方法和矩阵干预策略的有效性，在真实数据集 LSED 上的案例分析进一步展示了方法的实践价值.

未来工作将包括如下两方面，首先，当前只探索了曼哈顿距离和切比雪夫距离两种距离度量函数，然而，它们只是相对基础的度量方式，还有诸如马氏距离等能够考虑到数据分布特点的度量方式，计划未来继续探索，该方案将有助于进一步提升变点检测的准确率；其次，矩阵干预策略在精度要求较高的时候，性能提升较为有限，还有优化的空间，可以考虑度矩阵在时间上的关联性，或者研究可用于微调社区结构矩阵的其他客观先验条件. 再次，数字生态上的社区检测和变点检测可以结合在一起进行，共同提升两个研究任务的准确性，扩展方法的应用能力.

参　考　文　献

[1] Briscoe G, De Wilde P. Digital ecosystems: Evolving service-oriented architectures//Proceedings of the 2006 1st Bio-Inspired Models of Network, Information and Computing Systems. Madonna di Campiglio, Italy, 2006: 17-es

[2] Wang Li, Cheng Xue-Qi. Dynamic community in online social networks. Chinese Journal of Computers, 2015, 38(2): 219-237(in Chinese)
（王莉，程学旗. 在线社会网络的动态社区发现及演化. 计算机学报，2015，38(2)：219-237）

[3] Xue X, Li G, Zhou D, et al. Research roadmap of service ecosystems: A crowd intelligence perspective. International Journal of Crowd Science, 2022, 6(4): 195-222

[4] Xue X, Chen Z, Wang S, et al. Value entropy: A systematic evaluation model of service ecosystem evolution. IEEE Transactions on Services Computing, 2022, 15(4): 1760-1773

[5] Yan Bo, Liu Xiao-Feng. Research of ecological construction of data circulation and sharing in large group enterprises. Information Technology & Standardization, 2023, (12): 100-106(in Chinese)
(闫博, 刘晓峰. 大型集团型企业数据流通共享生态建设研究. 信息技术与标准化, 2023, (12): 100-106)

[6] Li Chuan-Quan, Fang Lan-Ran, Su Qi, et al. A research on the open source ecosystem based on complex networks: A case study for R language. Journal of Systems Science and Mathematical Sciences, 2023, 43(8): 1993-2012(in Chinese)
(李传权, 方岚然, 苏琦等. 基于复杂网络的开源软件生态系统研究——以 R 软件为例. 系统科学与数学, 2023, 43(8): 1993-2012)

[7] Liu Ming-Yi, Tu Zhi-Ying, Xu Xiao-Fei, et al. Multi-level service ecosystem evolution analysis based on stochastic block model. Chinese Journal of Computers, 2022, 45(4): 798-811(in Chinese)
(刘明义, 涂志莹, 徐晓飞等. 基于随机块模型的多层次服务生态系统演化分析. 计算机学报, 2022, 45(4): 798-811)

[8] Rong K, Hu G, Lin Y, et al. Understanding business ecosystem using a 6C framework in Internet-of-Things-based sectors. International Journal of Production Economics, 2015, 159: 41-55

[9] Xia Xiao-Ya, Zhao Sheng-Yu, Han Fan-Yu, et al. Data mining and information service for open collaboration digital ecosystem. Computer Science, 2024. https://link.cnki.net/urlid/50.1075.TP.20240318.2024.003(in Chinese)
(夏小雅, 赵生宇, 韩凡宇等. 面向开源协作数字生态的信息服务与数据挖掘. 计算机科学, 2024. https://link.cnki.net/urlid/50.1075.TP.20240318.2024.003)

[10] Liu M, Tu Z, Zhu Y, et al. Data correction and evolution analysis of the ProgrammableWeb service ecosystem. Journal of Systems and Software, 2021, 182: 111066

[11] Donnat C, Holmes S. Tracking network dynamics: A survey of distances and similarity metrics. arXiv, 2018. https://arxiv.org/abs/1801.07351

[12] Zheng Zuo-Wu, Shao Si-Qi, Gao Xiao-Feng, et al. Social circle and attention based information popularity prediction. Chinese Journal of Computers, 2021, 44(5): 921-936(in Chinese)
(郑作武, 邵斯绮, 高晓沨等. 基于社交圈层和注意力机制的信息热度预测. 计算机学报, 2021, 44(5): 921-936)

[13] Bongini P, Bianchini M, Scarselli F. Molecular generative graph neural networks for drug discovery. Neurocomputing, 2021, 450: 242-252

[14] Guo Z, Zhang C, Yu W, et al. Few-shot graph learning for molecular property prediction//Proceedings of the Web Conference 2021. Ljubljana, Slovenia, 2021: 2559-2567

[15] Chen J, Liu F, Jiang J, et al. TraceGra: A trace-based anomaly detection for microservice using graph deep learning. Computer Communications, 2023, 204: 109-117. doi: 10.1016/j.comcom.2023.03.028

[16] Chen H, Chu L. Graph-based change-point analysis. Annual Review of Statistics and Its Application, 2023, 10(1): 475-499

[17] Wang Y, Chakrabarti A, Sivakoff D, et al. Fast change point detection on dynamic social networks//Proceedings of the 26th International Joint Conference on Artificial Intelligence. Melbourne, Australia, 2017: 2992-2998

[18] Sulem D, Kenlay H, Cucuringu M, et al. Graph similarity learning for change-point detection in dynamic networks. arXiv preprint arXiv:2203.15470, 2022

[19] Xu C, Lee T C M. Statistical consistency for change point detection and community estimation in time-evolving dynamic networks. IEEE Transactions on Signal and Information Processing over Networks, 2022, 8: 215-227

[20] Miller H, Mokryn O. Size agnostic change point detection framework for evolving networks. PLoS One, 2020, 15(4): e0231035

[21] Zhu T, Li P, Yu L, et al. Change point detection in dynamic networks based on community identification. IEEE Transactions on Network Science and Engineering, 2020, 7(3): 2067-2077

[22] Cheng J, Chen M, Zhou M, et al. Overlapping community change-point detection in an evolving network. IEEE Transactions on Big Data, 2020, 6(1): 189-200

[23] Hewapathirana I U, Lee D, Moltchanova E, et al. Change detection in noisy dynamic networks: A spectral embedding approach. Social Network Analysis and Mining, 2020, 10(1): 14

[24] Barnett I, Onnela J P. Change point detection in correlation networks. Scientific Reports, 2016, 6(1): 18893

[25] Marenco B, Bermolen P, Fiori M, et al. Online change point detection for weighted and directed random dot product graphs. IEEE Transactions on Signal and Information Processing over Networks, 2022, 8: 144-159

[26] He Y, Kong X, Trapani L, et al. Online change-point detection for matrix-valued time series with latent two-way factor structure. arXiv preprint arXiv:2112.13479, 2021

[27] Rossetti G, Pappalardo L, Pedreschi D, et al. Tiles: An online algorithm for community discovery in dynamic social networks. Machine Learning, 2017, 106(8): 1213-1241

[28] Wu Yue, Wang Ying, Wang Xin, et al. Motif-based hypergraph convolution network for semi-supervised node classification on heterogeneous graph. Chinese Journal of Computers, 2021, 44(11): 2248-2260(in Chinese)
(吴越, 王英, 王鑫等. 基于超图卷积的异质网络半监督节点分类. 计算机学报, 2021, 44(11): 2248-2260)

[29] Hao J, Zhu W. Deep graph clustering with enhanced feature representations for community detection. Applied Intelligence, 2023, 53(2): 1336-1349

[30] Sserwadda A, Ozcan A, Yaslan Y. Structural and topological guided GCN for link prediction in temporal networks. Journal of Ambient Intelligence and Humanized Computing, 2023, 14(7): 9667-9675

[31] Hou X, Ma R, Yan L, et al. DAuCNet: Deep autoregressive framework for temporal link prediction combining copy mechanism network. Knowledge and Information Systems, 2023, 65(5): 2061-2085

[32] Cao M, Song J, Yuan J, et al. FairHELP: Fairness-aware heterogeneous information network embedding for link prediction//Proceedings of the International Conference on Database Systems for Advanced Applications. Tianjin, China, 2023: 320-330

[33] Ryck T D, Vos M D, Bertrand A. Change point detection in time series data using autoencoders with a time-invariant representation. IEEE Transactions on Signal Processing, 2021, 69: 3513-3524

[34] Zhu D, Ma Y, Liu Y. A flexible attentive temporal graph networks for anomaly detection in dynamic networks// Proceedings of the 2020 IEEE 19th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom). Guangzhou, China, 2020: 870-875

[35] Suárez J L, García S, Herrera F. A tutorial on distance metric learning: Mathematical foundations, algorithms, experimental analysis, prospects and challenges. Neurocomputing, 2021, 425: 300-322

[36] Zhu Ye-Qi, Liu Ming-Yi, Wang Zhong-Jie. A digital ecosystem evolution observation method and system based on continuous community boundary detection: CN202211252317. 2023-02-07 (in Chinese)
(朱业琪，刘明义，王忠杰. 一种基于持续社区边界检测的数字生态演化观测方法及系统：CN202211252317. 2023-02-07)

[37] Rossetti G. RDyn: Graph benchmark handling community dynamics. Journal of Complex Networks, 2017, 5(6): 893-912

[38] Huang S, Hitti Y, Rabusseau G, et al. Laplacian change point detection for dynamic graphs//Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. CA, USA, 2020: 349-358

ZHU Ye-Qi, Ph. D. candidate. His research interests include service computing, complex network analysis, graph neural networks, and data mining.

LIU Ming-Yi, Ph. D., assistant professor. His research interests include service ecosystem modeling, service evolution analysis, data mining, and graph neural networks.

SU Tong-Hua, Ph. D., professor. His research interests include pattern recognition, handwritten Chinese character recognition, deep learning, and heterogeneous computing.

WANG Zhong-Jie, Ph. D., professor. His research interests include services computing and software engineering.

Background

With the development of technologies such as Big Data, Internet of Things, and Cloud Computing, the representation of the digital ecosystem has become increasingly complex, interlacing with our lives more tightly. Many things show the concept of digital ecosystem, such as social network and service ecosystem. A Digital ecosystem is often a dynamic network. The evolution of a digital ecosystem can be observed through the network. The ongoing evolution of the digital ecosystem might generate unexpected relationships within. Under this scenario, change point detection is an effective approach to address the evolution analysis of a digital ecosystem. However, there are seldom studies that propose a method for change point detection that considers the characteristic of the digital ecosystem. The existing approaches may not support the detection in continuous scenario. In this case, the existing approaches cannot track the evolution continuously. They can only locate the causes of evolution to a batch of events. Besides, they may present performance degradation issues. Therefore, it is necessary to propose a change point detection approach for digital ecosystem that support the continuous-time dynamic graph modeling. This paper proposes MDCPD: Change Point Detection for Digital Ecosystem Based on Sequenced Matrices Distance Measurement, and the main contributions are as follows. (1) MDCPD leverages matrix distance measurement to determine whether a change point exists, which supports both continuous-time and discrete-time modeling. (2) Propose matrix intervention strategy, which reduces the noise in the matrices and improves the performance of MDCPD. (3) A series experiments are conducted on a dynamic dataset, including comparison with SOTA approaches, an ablation study on matrix interference strategy, and a case study conducted on a dataset from the real world, which suggests MDCPD is very practical.

Research in this paper is partially supported by the National Key Research and Development Program of China (No. 2021YFB3300700) and the National Natural Science Foundation of China (62372140, 62277011).