计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
2026,35(1):164−177 [doi: 10.15888/j.cnki.csa.010031] [CSTR: 32024.14.csa.010031]

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

基于活动案例间行为信息编码的时间预测①

张振涛[1], 方贤文[1,2]

[1](安徽理工大学 数学与大数据学院, 淮南 232001)
[2](安徽理工大学 安徽省煤矿安全大数据分析与预警技术工程实验室, 淮南 232001)
通信作者: 方贤文, E-mail: 280060673@qq.com

摘　要: 预测性过程监控 (predictive process monitoring, PPM) 技术通过分析已有的事件日志对正在运行的业务流程中的某些关键指标进行预测. 目前的预测性过程监控方法在特征提取方面通常预设案例仅受自身属性的影响或仅通过提取资源案例间行为属性进行编码, 通常未涉及活动视角下的案例间行为信息. 本文提出了一种捕获活动案例间行为信息的方法 IABC (inter-activity behaviour of case) , 该方法设计了一个三维特征构建框架涵盖时间窗口、活动粒度、行为状态这 3 大维度, 共构造 36 种活动案例间行为特征. 同时, 本文提出了两项关键算法: 影响力分布算法用于挖掘活动间的正/负影响传播; 批次行为检测算法用于识别潜在的批量操作行为. 最后在 3 个公开可用的事件日志上评估 IABC 方法的有效性, 结果表明集成了 IABC 方法的时间预测模型性能优于未使用该方法的基线模型和采用了资源案例间特征的模型.

关键词: Transformer; 业务流程预测; 活动行为信息; 案例间编码; 时间预测

引用格式: 张振涛,方贤文.基于活动案例间行为信息编码的时间预测.计算机系统应用,2026,35(1):164–177. http://www.c-s-a.org.cn/1003-3254/10031.html

Time Prediction Based on Inter-case Behavioral Information Encoding of Activities

ZHANG Zhen-Tao[1], FANG Xian-Wen[1,2]

[1](School of Mathematics and Big Data, Anhui University of Science & Technology, Huainan 232001, China)
[2](Anhui Province Engineering Laboratory for Big Data Analysis and Early Warning Technology of Coal Mine Safety, Anhui University of Science & Technology, Huainan 232001, China)

Abstract: Predictive process monitoring (PPM) techniques utilize existing event logs to predict certain key metrics in running business processes. In terms of feature extraction, current PPM methods presuppose that cases are solely influenced by their attributes or exclusively encoded by extracting resource-level inter-case behavioral attributes. These methods typically overlook inter-case behavioral information from the activity perspective. This study proposes a new method to capture the inter-activity behaviour of cases (IABC), which involves a feature construction framework covering three dimensions: time window, activity granularity, and behaviour state. It constructs a total of 36 types of inter-activity behavioral features. Concurrently, this study proposes two novel algorithms: the influence distribution algorithm for mining positive/negative influence propagation among activities, and the batch behaviour detection algorithm for identifying potential batch operations. The effectiveness of the IABC method is evaluated on three publicly available event logs. The results demonstrate that the temporal prediction model integrating the IABC method outperforms both the baseline model, which does not use the method, and the model that employs resource-level inter-case features.

Key words: Transformer; business process prediction; activity behaviour information; inter-case coding; time prediction

① 基金项目: 国家自然科学基金 (61572035); 安徽省重点研究与开发计划 (2022a05020005); 安徽省自然科学基金 (2308085US11)
收稿时间: 2025-05-27; 修改时间: 2025-06-24; 采用时间: 2025-07-14; csa 在线出版时间: 2025-10-21
CNKI 网络首发时间: 2025-10-22

业务流程的基础是企业软件系统 (客户关系管理系统 CRM、供应链管理系统 SCM 等) 或信息系统运行过程, 由个人或系统执行特定业务活动产生的结构化日志记录. 这些日志中包含了流程执行过程中相关事件的详细记录. 事件日志是轨迹的集合, 每条轨迹都捕获了一次特定业务活动执行期间涉及的所有事件, 事件按时间戳排序, 通常包括了活动、资源、完成状态等基于专业领域的必要属性.

预测性业务流程监控 (PBPM)[1]是一系列技术, 其利用业务流程系统产生的事件日志来生成系统中每个正在进行案例的未来状态或属性预测. 预测目标多种多样, 包括预测特定流程尚未执行的剩余轨迹和下一个活动、剩余时间和下一个活动时间、特定流程结果等. 为了实现这些预测, 通常根据预测目标训练相应的预测模型, 特征向量则是由事件日志中部分流程 (活动前缀) 形成的轨迹特征与其对应的事件属性形成的属性特征拼接得到[2]. 在运行阶段, 模型利用当前事件的特征向量生成对应的预测结果. 在除活动预测外的所有预测任务中, 活动前缀是已知的, 且编码方法是确定的, 所以在不改变模型结构与种类的情况下, 预测性能很大程度上取决于属性特征对日志信息的表示能力[3].

当前的主流分析方法通常将案例视为彼此独立执行的流程[4], 较少深入考虑案例之间在行为层面的互动关系——例如, 一个案例的执行状态或决策如何受到其他同时运行案例的影响. 虽然已有研究开始关注案例间的关系, 但这些工作主要探讨资源视角下的分配、争用、效率等问题[5,6]. 相比之下, 对于活动视角下的信息流传递、批次流影响等其他行为机制如何影响案例执行, 特别是如何将这些行为编码为可被模型使用的特征, 现有的探索仍显不足.

案例间行为信息是指案例之间由于竞争、协同等相互作用而产生的信息, 这些信息在单个流程实例中无法被察觉. 行为是流程对周围状态的反应, 当多个流程在业务流程系统中并行执行时就会产生复杂的状态, 在这种状态下便会产生复杂的高级行为, 类似批处理、延迟、周期执行等. 高级行为[7]存在且只存在于多个案例中, 其在单个流程中无法检测. 对于以时间或异常为预测目标的业务流程监控, 这些高级活动也是性能瓶颈或预测偏差产生的主要原因.

例如在一个顾客反馈的流程中, 存在约束每天固定时间点开始处理业务反馈, 这种约束会导致任务的积压与资源负载的动态变化, 由此产生了跨案例间的行为信息交互, 具体表现为资源处于高负载状态, 案例间产生高级行为 (批开始). 这类表现的本质上是跨案例交互 (资源争用或批量规则) 的结果, 需通过多案例分析提取并编码为事件特征, 才能在单案例模型中体现, 在资源视角与活动视角下二者之间并不等价.

在预测性流程监控中显式表达案例间行为信息能为性能瓶颈的识别与性能提升提供数据支持与特征筛选策略. 特别是对于各种预测模型, 显式的案例间行为信息能够提高模型对活动间依赖信息的捕捉能力, 从而提高预测的准确性.

对于案例间行为信息的提取与编码, 已有方法基于资源视角, 对业务流程中的资源负载、经验等进行统计表示, 但是这些方法忽略了活动视角下多流程运行过程中活动本身所产生的案例间行为信息. 因此, 本文提出了名为 IABC 的案例间行为信息捕获与编码方法, 该方法主要针对活动案例间行为信息. 本文的主要贡献如下.

(1) 提出了一种名为 IABC 的活动案例间行为的信息编码方法, 该方法包括影响力分布算法、批次行为检测算法, 结合传统的数据聚合方法能够从事件日志中提取案例间行为信息并编码为特征.

(2) 在 3 个公开可用数据集上进行仿真实验, 并对实验结果进行分析, 结果表明本文提出的方法可以有效提高业务流程时间预测的相关精度.

(3) 本文对 IABC 方法所构造的特征进行解释分析, 给出在业务流程时间预测方面对于活动案例间行为特征选择的具体建议.

1 相关工作

在业务流程预测性监控领域, 案例间行为的相互影响是一个备受关注的问题. 众多研究致力于解决多流程并发场景中案例间行为相互影响与其对整体系统性能的影响的问题. 在 2018 年业务流程管理国际会议 (BPM) 中, 文献[8]率先论证提出案例间行为相互影响的存在性, 同时性能表现会随时间发生非线性变化. 随后, 2020 年流程挖掘国际会议 (ICPM), 文献[9]进一步指出多个案例之间的相互影响是导致动态性能瓶颈出现的关键诱因, 对整体系统性能构成显著挑战. 文献[10]通过实验证明, 添加案例间行为特征可以提高几乎所有评估的主要预测方法在两个真实事件日志上的剩余

时间预测性能.

案例间行为信息的提取是一个综合性的复杂问题. 文献[11]提出了一种二维状态空间表示的特征编码框架, 用于同时编码案例内与案例间行为特征, 用于表示系统中对稀缺资源的争用. 同时该文献也确定了在案例间行为信息提取方面先确定特征维度后进行数据聚合的核心思路. 文献[12]确立了一个具体的特征维度框架, 详细建议了在时间窗口内所确立的案例间属性的类别.

在近 3 年的研究中, 文献[13]设计了一个用于资源经验的特征提取框架, 该框架通过捕获业务流程系统所涉及的资源的过往经验从而捕获资源视角下的案例间行为信息, 通过对信息编码为特征参与业务流程预测性监控. 文献[14]设计了一种名为 LS-ICE 的特征提取框架, 用于提取以资源负载为核心的案例间行为信息, 该方法通过将时间段内的资源负载状况编码为特征参与预测. 上述两种方法均为在资源视角下对案例间行为信息提取编码, 在活动视角下对案例间行为信息编码方面依旧存在空白, 但已经存在部分研究对活动视角下的高级行为进行定义与检测. 在 2022 年 ICPM 会议上, 文献[7]提出一种具体的框架用于捕捉无法在单个流程中体现的高级行为, 并生成高级事件日志. 同时作者在后续的工作 (文献[15]) 中给出了具体的高级行为定义与其行为关联机制, 确立了具体的高级行为捕获方法. 文献[16]则是将案例间行为定义为情景模式, 通过将案例进行拆分后检测的方式, 定义了一种在事件日志中检测出模式的方法.

截至目前, 现有的案例间行为信息提取具有以下两点局限性.

(1) 已存在活动视角下案例间高级行为的定义与检测, 但是不存在系统化的特征维度框架, 导致此类行为无法显式编码为结构化的预测特征.

(2) 现有预测性监控方案中, 活动视角的案例间交互信息没有被显式编码, 仅能依赖预测模型的长短期依赖关系隐式捕捉, 制约了预测模型对日志中信息的利用.

2 基础知识

2.1 基础定义

定义 1 (事件、属性). 事件实例e是一个元组, 其形式表达为: $e=(c,a,t,r,(d_1,v_1),\cdots,(d_m,v_m))$. c是事件所属的业务流程实例唯一标识; a是活动标签; t是事件e对应的时间戳; r是执行该事件的资源; (d_m,v_m)则用于表达与事件相关的属性和其对应的值. 例如, 事件$e_2=(63,\text{approve},2020/3/15,\text{Bob},5000,\text{fast})$捕获了这样一个事实: 在与第 63 个贷款请求关联的流程案例中, 资源 Bob 在 2020 年 3 月 15 日对 5 000 美元的贷款请求执行了快速审批. 值得注意的是本文对于提出的事件只需要基础属性c、a、r. 此处使用E表示所有事件的集合, A、R分别表示所有活动和所有资源的集合, T表示所有时间戳的集合. 本文使用点状表示法访问事件对应的属性. 例如$e.r$表示事件e的对应的资源属性r, 示例事件日志如表 1 所示.

表 1 事件日志示例

事件	案例	活动	时间	资源	amount	type
e_1	63	submit_request	2020/3/10	Alice	5 000	standard
e_2	63	manual_review	2020/3/12	Carol	5 000	standard
e_3	63	approve	2020/3/15	Bob	5 000	fast
e_4	84	submit_request	2020/4/1	Alice	10 000	fast
e_5	84	reject	2020/4/5	Dave	10 000	standard

定义 2 (轨迹、事件日志). 在给定情况下生成的事件序列$\sigma=\langle e_1,e_2,\cdots,e_n\rangle$需满足$\forall e_i,e_j\in\sigma,\ i<j\in[1,n]$; $e_i.c=e_j.c\wedge e_i.t<e_j.t$, 则称$\sigma$是一条轨迹, 轨迹内的事件具有相同的案例标识符且按时间戳排序, n代表轨迹长度. 事件日志定义为一组轨迹的集合$L=\langle\sigma^1,\sigma^2,\cdots,\sigma^m\rangle$. 其中$m$是事件日志$L$中的轨迹总数, σ^m表示事件日志L中的第m个轨迹.

定义 3 (事件级属性、案例级属性). 事件级属性是相对案例级属性而言. 案例级属性即对于轨迹σ^m的所有事件e的某个属性满足$\forall e\in\sigma^m: e.d=v$, 其中 v 是常数, 则称该属性为案例级属性, 如果不是则是事件级属性.

定义 4 (轨迹前缀、案例活动状态、全局活动状态). 给定一个长度为n的轨迹σ与正整数i $(i\leqslant n)$, 函数$prefix(\sigma,i)$用于捕获σ的前i个事件对应的活动标签, 即$prefix(\sigma,i)=(e_1.a,\cdots,e_i.a)$. 函数$case_state(\sigma,i)$用于捕获轨迹$\sigma$内第$i$个事件对应轨迹的前后事件的活动标签, 即$case_state(\sigma,i)=(e_{i-1}.a,e_{i+1}.a)$, 表示案例活动状态. 函数$totalstate(e_l.t,L)$用于捕获事件$e_l$在事件日志内前后事件对应的活动标签即$totalstate(e_l.t,L)=(e_{l-1}.a,e_{l+1}.a)$. 其中$e_{l-1}$满足$\forall e\in L,\ e\neq e_{l-1},\ e.t<e_l.t: |e.t-e_l.t|\geqslant|e_{l-1}.t-e_l.t|$. 同理, e_{l+1}满足$\forall e\in L,\ e\neq e_{l+1},\ e.t>e_l.t: |e.t-e_l.t|\geqslant|e_{l+1}.t-e_l.t|$.

定义 5 (交接、任务). 给定轨迹中两个事件如果满足 $e_i, e_j\ (j-i=1): e_i.c = e_j.c \wedge e_i.r \neq e_j.r$, 则认为这两个事件间存在一个交接, 使用 $e_i.a \rightarrow e_j.a$ 表示; 即交接用于表达某轨迹中资源的转换. 事件序列 $\sigma^n = \langle e_1, e_2, \cdots, e_m \rangle$ 被定义为任务需要满足 $\forall e_i, e_j \in \sigma^n,\ j-i=1: e_i.c = e_j.c \wedge \neg \exists e_i.a \rightarrow e_j.a$. 任务用于表示某资源执行的一系列事件, 与事件序列和资源相关.

定义 6 (时间窗口). $\forall e_i.t, e_j.t \in T: e_i.t < e_j.t$ 则称其形成了一个时间窗口 $\left(e_i.t, e_j.t\right)$, 本文涉及的时间窗口具体划分如图 1 所示.

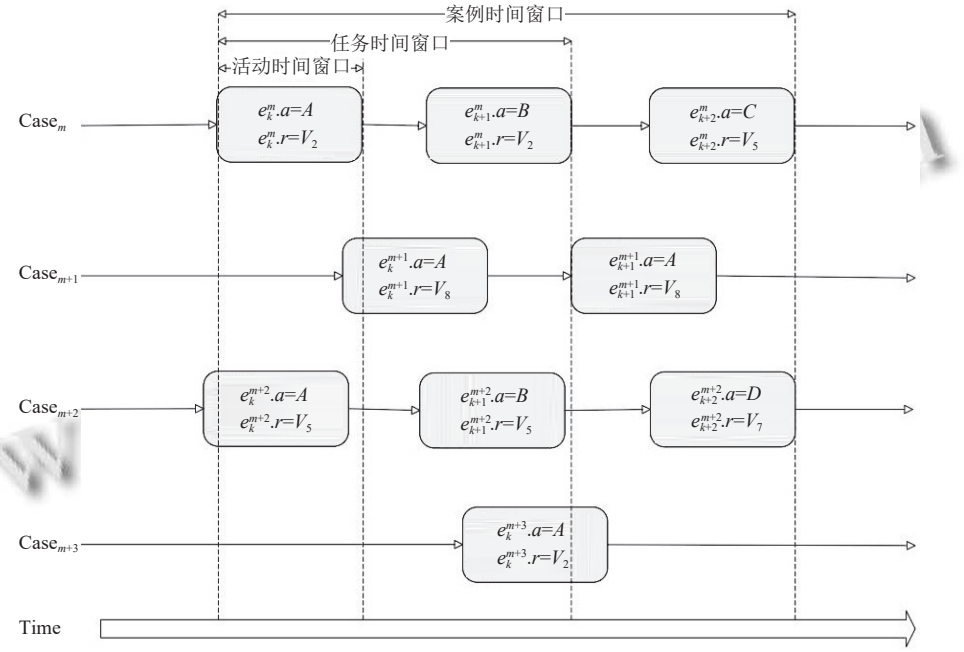


图 1 时间窗口划分

3 方法

3.1 业务流程监控 Transformer 时间预测模型

本文所使用的模型是在传统的 Transformer 模型上改进得到的, 引用了已有 Transformer 模型[17]的设计. 因为预测性流程监控中的时间预测不涉及对序列的重新排列, 所以本文将传统 Transformer 模型中的解码层替换为全连接层, 同时为了避免过拟合, 本文所使用的模型多次引入 Dropout 层. 该时间预测框架通过将本文中所提出的 IABC 方法得到的案例间活动行为特征与轨迹前缀特征, 分别输入模型以丰富当前事件的特征. 在对事件通过 IABC 方法丰富之后, 选取单特征与轨迹前缀一同送入 Transformer 模型中进行训练与预测, 本文对于剩余时间预测与下一个活动时间预测两种不同的任务, 在模型结构上没有太大的差别. 具体的 Transformer 模型结构如图 2 所示.

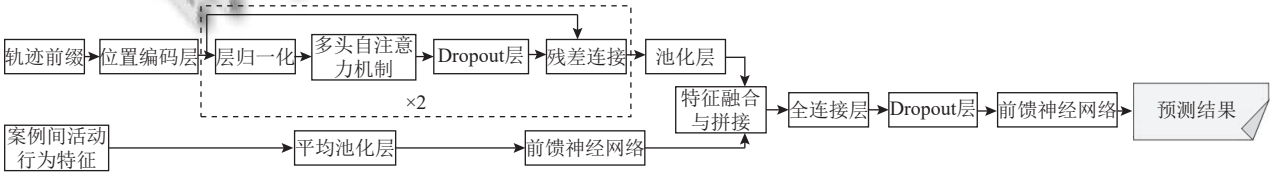


图 2 业务流程中用于时间预测的 Transformer 模型

3.2 IABC 方法

案例从来不是单独运行的, 而是在业务流程系统中并行执行. 为了构建训练模型相关的特征, 本文提出 IABC 方法用于提取多流程并发状态下的案例间行为信息交互, 具体模型框架如图 3 所示. 原始的事件日志会通过 IABC 方法中的算法部分对日志进行粗加工, 其中影响力分布算法与批次行为检测算法会分别生成活动对影响力分布矩阵与批次 ID 列, 同时根据所提取

活动案例间行为信息不同, IABC 方法会根据 3 个不同的维度确定对应的特征, 构造形成对应的案例间行为信息的特征. 结合粗处理后的事件日志与构造的特征就可以从原始事件日志中编码出所需的案例间行为信息特征. 此时 IABC 方法所做的主要工作已经完成, 后续工作会将该特征与活动前缀轨迹特征同时送入时间预测 Transformer 模型进行训练与预测. 对于具体的 IABC 方法, 本节首先介绍具体的 3 个维度, 在介绍最后一个维度时会介绍 IABC 方法中对事件日志进行粗处理的两个算法.

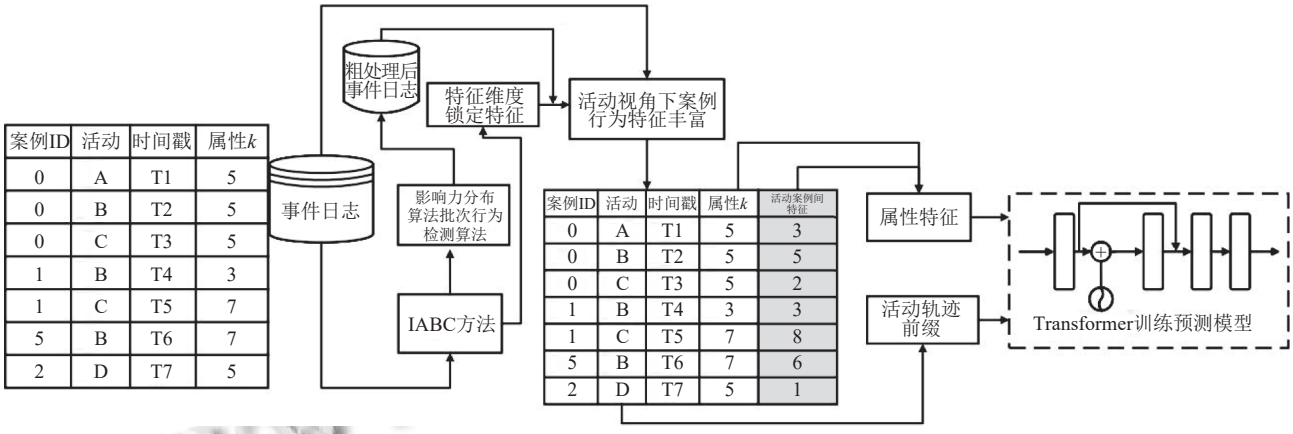


图 3 IABC 方法总体框架

本文目标是提取活动案例间行为信息, IABC 方法将这一目标表述为提取某段时间内与活动相关对象的行为状态表示. 因此我们将案例间行为信息表述为 3 个维度: 时间窗口、活动粒度、行为状态, 通过这 3 个维度的不同组合得到具体的特征目标, 具体维度划分如图 4 所示. 这种表征方法是对文献[7]中“高级活动”概念的拓展, 其中活动对象的概念与文献[18]中以对象为中心的流程挖掘中考虑的对象概念一致, 同样符合文献[19]中对事件日志粗粒度的抽象. 而时间窗口的概念则对应文献[15]中“段”的概念. 通过组合 3 个维度, 我们总计构建了 36 个特征, 这些特征能够帮助我们从不同角度提取活动案例间的行为信息. 接下来, 我们将详细讨论 IABC 方法具体关联的维度, 同时对每个维度, 确定其下属的分支值.

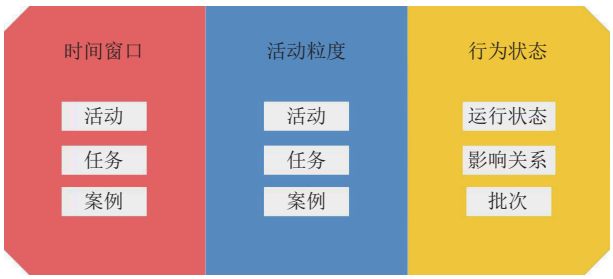


图 4 IABC 方法特征维度

3.3 IABC 的特征维度与算法

● 时间窗口: 本文中所提及的时间窗口具有 3 个等级, 活动级 (事件级)、任务级、案例级, 具体划分方法如图 1 所示, 在图中给出了基于案例 m 的第 k 个事件与第 k+1 个事件进行特征丰富时的时间窗口划分示例, 活动级时间窗口的范围是所关注事件从开始到结束的时间范围, 同时如果前后事件间不存在交接则满足任务的定义. 所以对于一个任务中的任意一个事件, 任务级时间窗口的范围是其所在任务的时间范围. 与任务级时间窗口同理, 案例级时间窗口的范围是其所在案例的时间范围. 这种时间窗口的划分方法可以完整地从活动、任务 (部分活动序列) 案例的角度分析时间段内所发生的案例间行为交互.

● 活动粒度: 本文中所提及的活动粒度分为 3 种, 分别是活动、任务、案例. 这一维度是指当前所关心的活动对象, 其划分的主要因素是对事件日志中不同层级的抽象. 活动是事件日志中的原子操作, 更是构成其他工作的原子结构, 所以是最先保留的活动对象. 之后在活动的基础上对事件日志进行抽象得到任务, 任务是在资源基础上对关联活动的聚合, 其保证了活动的部分连续性和案例间属性的提取. 最后是在案例层级上抽象事件日志得到案例. 在本文中, 我们将案例视为任务的聚合, 将其作为活动粒度的最高层级. 基于以

上粒度的划分, 另一个原因是对于资源与活动的组合考虑. 在事件日志中, 资源信息仅与单个工作项相关联, 所以活动层级保证了资源与活动直接关联, 同样对于任务层级, 其要求每个任务中不存在交接, 这种定义保证了在以资源作为案例间属性时, 其与轨迹中的部分活动直接关联. 而案例层级通过任务之间的顺序性保证了活动之间的联系. 简而言之, 我们通过资源这一案例间属性直接链接了所有活动, 任务是同一资源对于不同活动的链接, 而案例则是同一流程对于不同任务之间的链接.

● 行为状态: 在这一维度中, 行为状态的衡量分为 3 个方面, 分别是运行状态、影响关系和批次, 其中运行状态可以直接通过数据聚合得到, 而影响关系与批次两个方面则需通过本文所提出的两项算法对事件日志进行粗处理后得到基本信息再进行数据聚合才能够得到.

(1) 运行状态, 文献[12]中给出了运行状态涉及的特征, 但其重点放在了资源上. 在活动案例间行为信息提取方面, 我们关注的是某个具体的活动对象, 其在系统中所有同名活动的运行状态. 探究的是活动案例间行为是否可能存在, 及对于当前所关注活动对象的影响, 即对于某个活动对象, 关注系统中所有同类型活动的运行状态. 具体而言是关注其在一段时间内是否有同类型活动对象大量开始或结束, 值得注意的是, 我们没有要求统计活动级时间窗口下正在运行的活动数量, 主要基于以下两点: 1) 要求过于苛刻. 初步统计发现, 无论时间窗口如何选取, 关注活动级对象时该特征出现大量的 0 值, 没有统计意义; 2) 行为状态维度的批次一定程度弥补了对于某时间窗口内正在运行的活动的数量这一信息的提取. 综上所述, 我们为活动的行为状态给出了以下定义:

• 开始: 活动对象进入业务流程系统, 开始运行.

• 运行中: 活动对象在业务流程系统中正在执行.

• 结束: 活动对象离开业务流程系统, 结束运行.

(2) 影响关系: 在这个方面中主要捕捉活动对之间的单调关系 (只在活动层级捕捉), 即某个活动的执行时间是否会对其他活动的执行时间具有影响关系. 据此我们使用了 Spearman 相关系数衡量活动执行时间的单调关系, 因为该系数对分布与异常值并不敏感, 同时可以探究非线性关系. 需要强调的是得到的结果只是统计结果, 无法仔细探究其中具有的因果关系, 使用 Spearman 相关系数的算法如算法 1 所示.

算法 1. 影响力分布算法

输入: 原始事件日志 L.
输出: 影响力分布矩阵 M.

```
1. //根据活动类别对事件日志分组并提取执行时间
2. Groups←GroupByActivuty(L)
3. for G in Groups do
4.    //对每个组组内按时间戳升序排列
5.    SortByTs(G)
6. end for
7. //对每个组计算其与其他组的相关系数
8. for G_i in Groups do
9.    for G_j in Groups do
10.      //对齐两个分组至组内最短长度
11.      [G'_i,G'_j]←AlignByShortest(G'_i,G'_j)
12.      //计算 Spearman 相关系数作为影响力
13.      M[i][j]←SpearmanSim(G'_i,G'_j)
14.   end for
15. //输出影响力分布矩阵
16. end for
17. return M
```

算法 1 的输入是完整的事件日志, 输出则是由事件日志 L 得到的对应的影响力分布矩阵 M. 该算法首先对事件日志按照活动分组并提取执行时间, 第 2 步对每个组的组内按时间戳升序排列形成序列. 第 3 步对每个组计算其与其他各组之间的 Spearman 相关系数; 具体为将两组截取序列至相同长度 (两组之间的最短长度) 后进行计算. 最后统计所有计算结果就得到了影响力分布矩阵. 在得到影响力分布矩阵之后, 我们统计对于某个活动在对应时间段内受到其他活动具有加速影响、减速影响的活动的数量. 由此给出以下定义.

• 加速活动数: 具有加速影响的活动数.

• 减速活动数: 具有减速影响的活动.

(3) 批次: 在这个方面, 我们主要想了解的是活动之中是否可能存在批次关系, 同时是否存在多个批次, 每个批次的最大容量是多少. 在这个行为状态方面同时涉及资源与活动, 但关注的重点是活动之间的关系. 由此给出算法 2.

算法 2. 批次行为检测算法

输入: 原始事件日志 L.
输出: 批次 ID 列表.

```
1. //根据资源与活动类别进行分组
2. Groups←GroupEventsByAcRe(L)
3. //初始化全局批次 ID 列表与计数器
4. GlobalBatchIDList←[ ]
5. BatchCounter←0
```

```
6. for G in Groups do
7.   //对每个组组内按事件 ID 升序排序
8.   SortByEventsID(G)
9.   for i=1 to |G| do //对于组内的每一个事件
10.    e←G[i] //提取事件
11.    if i=1 then //每一组默认至少一个批次
12.      BatchCounter←BatchCounter+1
13.      e.batch←BatchCounter
14.    else //每组的非第 1 个事件
15.      prev←G[i–1] //当前事件上一个事件
16.      //当前事件目标时间数量级
17.      mag_e←⌊log10(e.startime)⌋
18.      //前一事件目标事件数量级
19.      mag_p←⌊log10(prev.startime)⌋
20.      //当前事件目标时间首位数字
21.      lead_e←⌊e.startime/10^mag_e⌋
22.      //上一事件目标时间首位数字
23.      lead_p←⌊prev.startime/10^mag_p⌋
24.      //同一数量级, 同一首位数, 目标时间递减
25.      if (mag_e=mag_p)∧(lead_e=lead_p)∧(e.startime⩽prev.startime) then
26.        e.batch←prev.batch //二者属于同一批次
27.      else //不满足则创建新批次
28.        BatchCounter←BatchCounter+1
29.        e.batch←BatchCounter
30.      end if
31.    end if
32.    Append(GlobalBatchIDList, e.batch) //记录 ID
33.  end for
34. end for
35. return GlobalBatchIDList //返回批次 ID 列表
```

算法 2 的输入是完整的事件日志, 输出则是事件日志 L 中对应的批次 ID 列表. 该算法首先对事件日志按活动与资源对进行分组, 同时对每个组内事件按事件 ID 升序排序; 第 2 步为组内事件确定批次, 遵循以下规则: 1) 每一组至少存在一个批次. 2) 如果某个非组内首事件目标时间的科学记数法满足数量级、数位数字都相同, 且其目标时间小于等于上一个事件的目标时间, 则其与上一个事件属于同一批次. 在使用算法进行处理后就会得到某个事件所属的批次 ID, 依据此结果我们为活动的批次分布维度给出了以下定义.

• 批次数量: 活动的批次的总数.

• 最大批次数: 批次包含的最大的活动数.

3.4 IABC 的活动维度特征编码

● 运行状态方面: 表 2 展示了在运行状态方面感知的活动案例间行为信息的属性特征, 其主要是活动维度与时间窗口、运行状态的组合, 用于计算指定时间窗口内活动、任务及案例的开始数、结束数与运行中数量. 该特征同时限制了活动属性, 即统计的活动对象都与特定的活动相关, 例如任务和案例都包含了指定的活动. 此处展示的频率方面的特征因为篇幅的原因没有显示时间窗口, 所以运行状态方面的实际特征总数是 24 个.

● 影响关系方面: 表 3 展示的是在影响关系方面感知的活动案例间行为信息的属性特征, 这一部分的主要目标是探究在给定时间窗口内活动与活动的相互影响. 又因为任务与案例的复杂性, 且并非原子工作项, 所以任务与案例的影响力关系是依赖于活动的影响力关系建立的, 即任务与案例的影响力关系是在该时间窗口内的活动的影响关系对应的数量作为特征. 这部分活动案例间行为信息属性特征涉及时间窗口与影响力关系, 总计 6 个特征.

表 2 运行状态方面案例间行为信息属性特征

特征序号	特征	状态	活动粒度	定义
0	num_events_start	开始	活动	e对应的活动开始的数量
1	num_tasks_start		任务	e对应的任务开始的数量
2	num_case_start		案例	e对应的案例开始的数量
3	num_tasks_running	运行	任务	e对应的活动运行中的数量
4	num_case_running		案例	e对应的案例运行中的数量
5	num_events_end	结束	活动	e对应的活动结束的数量
6	num_tasks_end		任务	e对应的任务结束的数量
7	num_case_end		案例	e对应的案例结束的数量

● 批次方面: 表 4 展示了在批次方面感知的活动案例间行为信息的属性特征. 这部分的特征不再只是与活动相关, 同时也涉及资源属性, 是对两方面的综合考虑. 这部分特征的主要目的是弥补和补充运行状态方面没有捕获的批次行为关系, 同时也为了提取案例间行为信息方面最为重视的批处理关系. 这部分活动案例间行为信息属性特征涉及时间窗口与批次, 总计 6 个特征.

表 3 影响关系方面案例间行为信息属性特征

特征序号	特征	时间窗口	定义
24	act_related_slow_down	活动	e对应事件级时间窗口内对$e.a$具有加速作用的事件的数量
25	act_related_speed_up		e对应事件级时间窗口内对$e.a$具有减速作用的事件的数量
26	task_related_slow_down	任务	e对应任务级时间窗口内对$e.a$具有加速作用的事件的数量
27	task_related_speed_up		e对应任务级时间窗口内对$e.a$具有减速作用的事件的数量
28	case_related_slow_down	案例	e对应案例级时间窗口内对$e.a$具有加速作用的事件的数量
29	case_related_speed_up		e对应事件级时间窗口内对$e.a$具有减速作用的事件的数量

表 4 批次方面案例间行为信息属性特征

特征序号	特征	时间窗口	定义
30	act_max_num_batch	活动	e对应活动级时间窗口内与$e.r$相关的批次的最大容量
31	act_num_batch		e对应活动级时间窗口内与$e.r$相关的批次的数量
32	task_max_num_batch	任务	e对应任务级时间窗口内与$e.r$相关的批次的最大容量
33	task_num_batch		e对应任务级时间窗口内与$e.r$相关的批次的数量
34	case_max_num_batch	案例	e对应案例级时间窗口内与$e.r$相关的批次的最大容量
35	case_num_batch		e对应案例级时间窗口内与$e.r$相关的批次的数量

IABC 方法通过组合 3 个维度, 总共构造了 36 个特征, 涉及 3 种不同的时间窗口, 3 种不同的活动粒度, 7 种具体的行为状态. 这些特征旨在支持现实流程中的瞬时性能瓶颈识别, 同时能够为优化无顺序约束的活动提供调度策略, 并为资源在批量操作场景下的负载状态与效率提供具体的数据支持. 这些特征提供的信息主要服务于性能优化与瓶颈识别的现实业务核心问题上. 在后续的实验部分, 因为特征数量的原因, 将会采用活动前缀加单特征的方式统一测试特征预测效果.

4 实验评估

4.1 实验设置与数据集

为验证本文方法对活动案例间行为信息捕获的适用性和有效性, 以及其在不同场景下的通用性, 本节设计了两组实验. 第 1 组实验旨在对比分析本方法与基线方法[17]与资源案例间信息方法[13]在不同业务流程领域事件日志上的具体应用效果; 第 2 组实验侧重于描述本方法对案例间行为信息特征之间的性能选择. 本文方法的核心目标在于捕获事件日志中的活动案例间行为信息, 评价指标是对于业务流程实例的下一个活动时间预测与剩余时间预测的精度指标 *MAE* 与 *RMSE*. 因此本节需要分别对这两项任务进行评估.

针对业务流程中的时间预测, 实验遵循标准的回归指标: 平均绝对误差 (*MAE*) 和均方根误差 (*RMSE*) 作为评估指标. *MAE* 与 *RMSE* 的定义形式如式 (1)、式 (2) 所示, 其中 n 是事件数量, y_i 是模型预测结果, $\hat{y}_i$ 是真实结果.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|(y_i - \hat{y}_i)| \tag{1}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{2}$$

本文实验使用了 3 个公开可用的数据集, 数据集具体信息见表 5, 所有事件日志都在 https://data.4tu.nl/ 上公开提供. bpic2015 数据集由荷兰 5 个城市提供, 该数据包含约 4 年内的所有建筑许可申请. 本实验使用了 5 个数据集中的第 2 个 (记为 bpic2015_2); helpdesk 数据集来自一家公司技术支持系统的问题处理过程; sepsis 数据集来自一家医院中脓毒症患者的处理流程. 在实验数据集的划分方面均采用 8:2 的比例划分训练集与测试集.

表 5 实验数据集信息

数据集	案例数量	事件数量	活动数量	最大轨迹长度
helpdesk	4 580	21 348	14	15
bpic2015_2	832	44 354	410	132
sepsis	1 050	14 920	15	185

本文采用的基础模型是 Transformer[17], 其注意力头数设定为 4, 表 6 提供了有关模型配置的详细信息, 以此进行网格搜索来确定模型超参数的适当值. 对于所有的事件日志与特征向量均采用相同的超参数网格搜索, 以在训练集上产生最低验证损失的超参数值组合认为模型达到了在该数据集与特征向量上的最佳结果, 之后使用取得最佳结果模型对测试集验证. 需要注意的是, 在模型的学习率部分同样采用了学习率调整器, 以 0.5 为衰变率, 5 为耐心参数对学习率进行动态调整.

表 6 超参数搜索网格

参数名称	参数搜索值
批次数量	32, 64, 128
学习率	0.001, 0.002, 0.004
早停参数	25, 50, 75
交叉验证数	2, 3, 5

4.2 实验结果

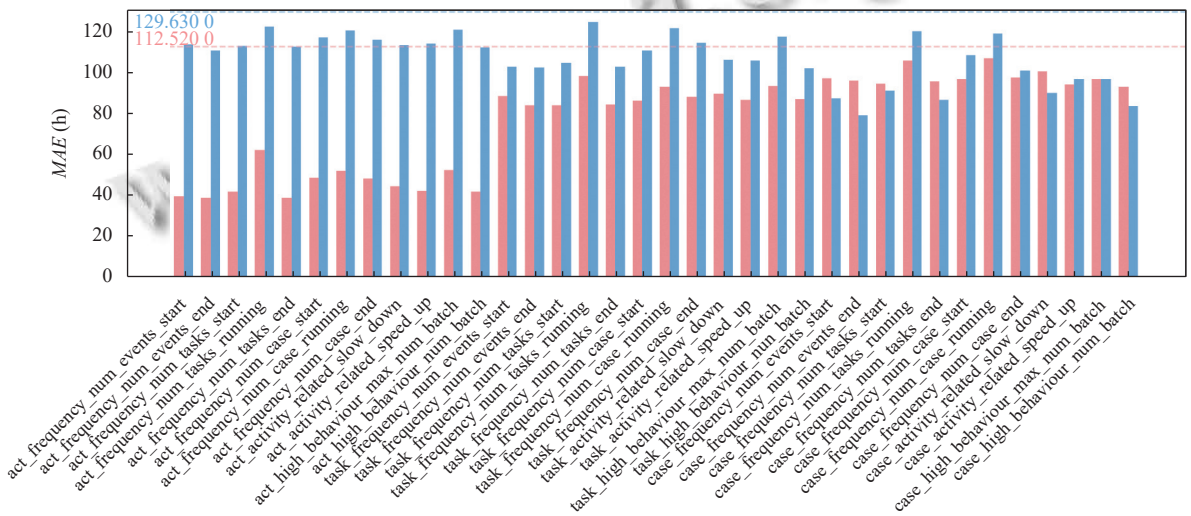
4.2.1 有效性评估

为验证本文所提方法在活动案例间信息提取方面的有效性, 本文在 3 个数据集上各自验证了 36 个活动案例间行为特征, 以及 27 个案例间资源特征在单特征预测下的综合时间预测结果. 资源特征的构造方法遵从文献[13]提及的方法, 评估指标选用使用平均绝对误差 *MAE* 和均方根误差 *RMSE* (以 h 为单位), 具体对比结果如表 7 所示, 其中的结果是综合了两种时间预测的总体体现, 其偏差主要来源于剩余时间预测方面.

表 7 综合有效性评估

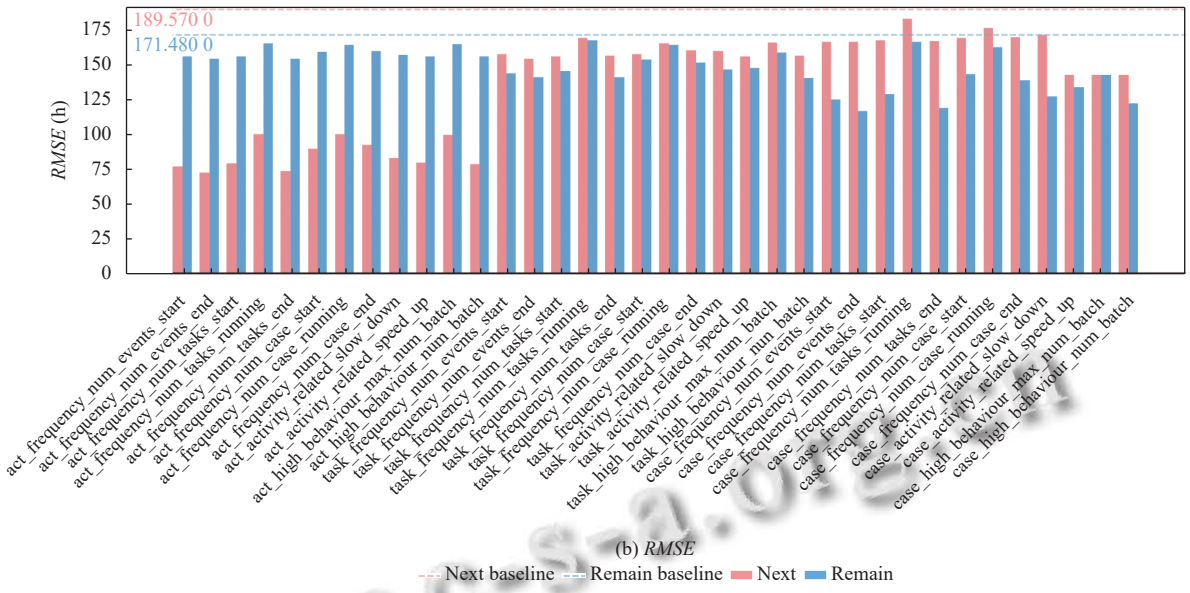
数据集	基线指标 (h)	资源 ($\mu\pm\sigma$)	活动 ($\mu\pm\sigma$)	相对基线[17]改进率 (%)	相对资源[13]改进率 (%)
helpdesk	*MAE* (121.07)	114.24±16.43	92.55±24.14	23.6	19.0
	RMSE (180.52)	172.01±13.84	143.19±29.67	20.7	16.8
bpic2015_2	*MAE* (1 595.82)	1399.20±1246.08	1178.90±1122.02	26.1	15.7
	RMSE (3 043.85)	2555.78±1777.50	2132.43±1699.48	29.9	16.6
sepsis	*MAE* (56.55)	51.04±42.28	47.00±40.35	16.9	7.9
	RMSE (83.58)	76.25±55.77	68.92±53.17	17.5	9.6

从表 7 中可以看出本文方法所捕获的活动案例间行为特征的总体表现优于基线方法[17]和资源案例间行为信息[13]捕获的特征. 其中在 helpdesk 数据集上明显优于资源案例间行为信息为特征的模型结构, 而在 bpic2015_2 上则相对基线方法有较大改善. 表 7 同时考虑两种时间预测任务的综合结果, 其较大的偏差范围主要源自剩余时间预测. 从结果与基线方法和资源方法的对比上来看, 本文提出的 IABC 方法有效降低了预测误差, 提高了准确性. 同时 IABC 方法在不同数据集上表现的误差增长趋势与数据集的复杂度成正比. 其在不同数据集上的具体表现与基线方法的对比结果如图 5–图 7 所示. 其中红色代表下一个活动时间预测的结果, 蓝色是剩余时间预测的结果. 在数据集上的具体表现而言, 活动数量最低, 轨迹长度最短的 helpdesk 数据集上的表现效果最好, 两种任务之间的差距不大, 这是由于活动数量与最大轨迹长度都不大, 这种业务流程系统中的案例间行为信息更加容易被捕捉到. 而对于 bpic2015_2 与 sepsis 数据集, 最大轨迹长度决定了两种不同时间预测任务的差距, 而活动的数量则决定了案例间行为信息的复杂程度, 实验结果显示的预测趋势符合正常业务流程时间预测结果的表现.



(a) *MAE*

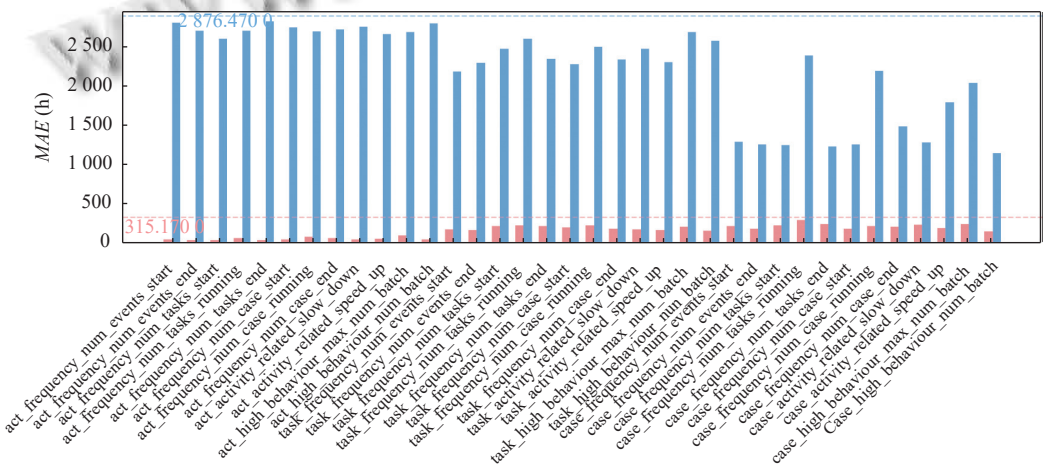
图 5 helpdesk 数据集的 IABC 方法结果



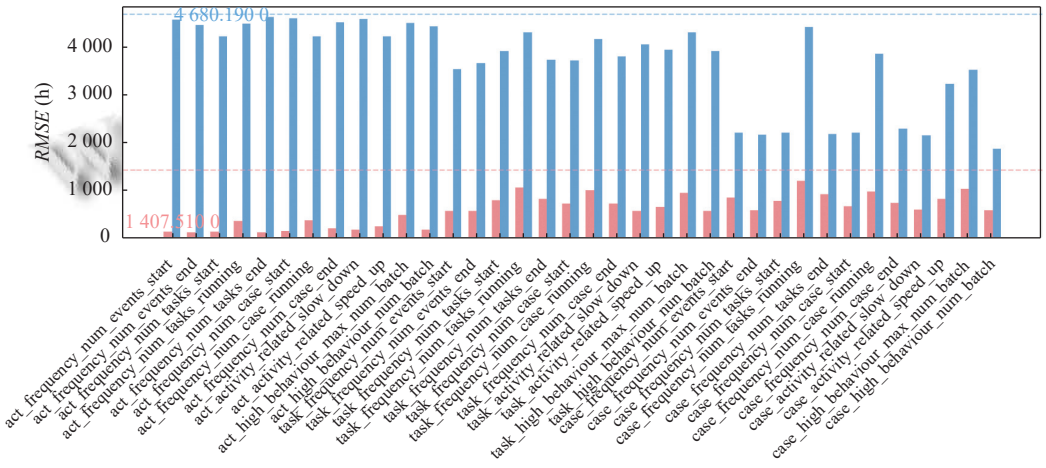
(b) *RMSE*

Next baseline Remain baseline Next Remain

图 5 helpdesk 数据集的 IABC 方法结果 (续)



(a) *MAE*



(b) *RMSE*

Next baseline Remain baseline Next Remain

图 6 bpic2015_2 数据集的 IABC 方法结果

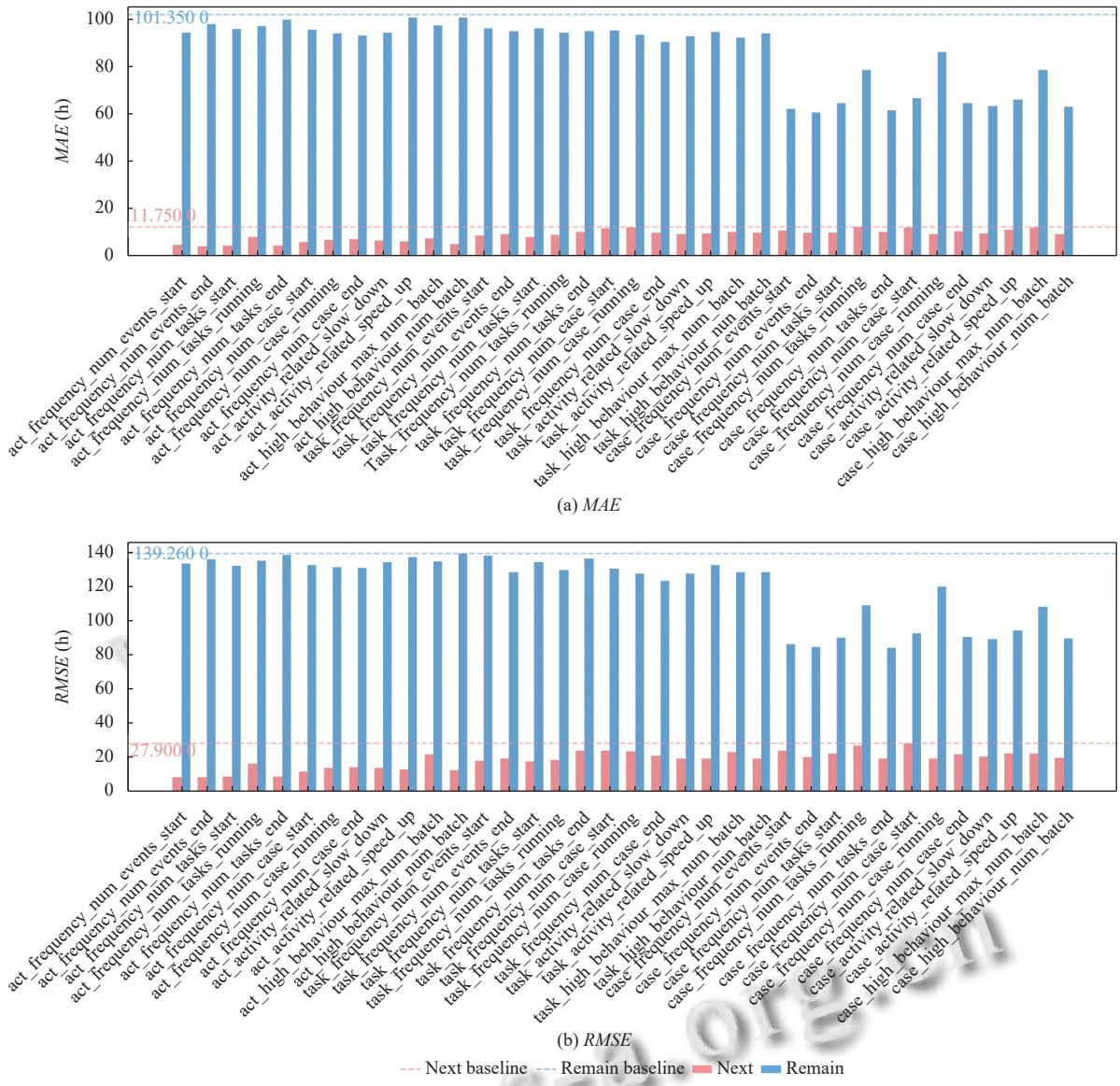


图 7 sepsis 数据集的 IABC 方法结果

4.2.2 特征性能评估

IABC 方法的活动案例间活动行为特征的构造由 3 个不同的维度共同影响, 对于两个不同的任务, 下一个活动时间预测与剩余时间预测的表现各维度的贡献各不相同, 这点从图 5–图 7 中也可具体体现. 对于下一个活动时间预测任务, 误差随着时间窗口增大而增大, 表现最好的行为状态的批次维度. 而对于剩余时间预测任务, 时间窗口的选择则与误差呈现反比, 较大的时间窗口能捕获更多的案例间活动行为信息, 其中表现最好的结果则产生于影响关系与批次维度.

对于具体的维度内构造的特征而言, 实验结果发现运行状态维度与常规思维相违背. 在 3 个数据集中判断当前正在运行的活动的数量即负载状态的数量所构造的特征的表现, 并没有比数据聚合方法下, 由活动开始与结束产生的案例间活动行为特征表现得更好. 同样, 对于影响关系维度, 加速事件数特征没有延缓事件数特征表现得更好. 这是由于在业务流程系统中, 更多的案例间活动行为信息表现为动态性能瓶颈或者延迟, 这种表现也符合早期文献[9]中的研究结果. 对于批次维度, 最大批次活动数没有批次数量表现得更好也是因为批次数量特征能够更加准确地描述案例间行为信息.

具体探究对于不同的任务应该具体使用哪些维度, 结果如图 8 所示, 图 8 从上到下依次为对应 *MAE* 与 *RMSE*, 从左到右依次为下一个活动时间预测与剩余活动时间预测. 结合图 8 的热力图表现, 我们会发现在下

一个活动时间预测任务中对于活动案例间行为信息的依赖主要集中在活动级时间窗口下, 行为状态维度的运行状态与批次两个方面. 随着前缀轨迹长度的增加和活动数量的增大, 运行状态维度的作用在上升. 在剩余时间预测任务中, 对于活动案例间行为信息的依赖, 主要体现在案例级时间窗口下的影响关系方面与批次方面, 其中影响关系能够在较大的轨迹长度下具有更好的表现, 而随着活动数量的增多, 批次维度的作用也开始上升. 这些数据都说明了本文所提影响力分布算法与批次行为检测算法都是真实有效的. 综上所述, 对于下一个活动时间预测, 时间窗口应该关注当前活动, 对于轨迹长度与活动数量都不大的小型数据集特征选择, 优先选择批次维度下的批次数量特征, 对于复杂的数据集可以考虑引入运行状态维度下的开始活动数量特征与结束活动数量特征. 而对于剩余时间预测任务, 优先选择影响关系维度下的减速活动数量特征与批次维度下的批次数量特征. 总而言之, 我们可以做出以下结论, 本文提出的 IABC 方法确实较为准确地捕捉到活动案例间行为信息.

以上实验结果为后续的特征筛选工作与在线业务流程监控系统的搭建, 提供了数据支持与优化策略. 图 5–图 7 的实验结果为后续数据驱动的特征筛选工作提供了依据, 同时图 8 所揭示的维度敏感性规律将为在线业务流程监控系统的开发提供核心配置模板, 通过预置高价值维度组合, 该系统能够更加精准地展示当前业务流程系统的性能状态.

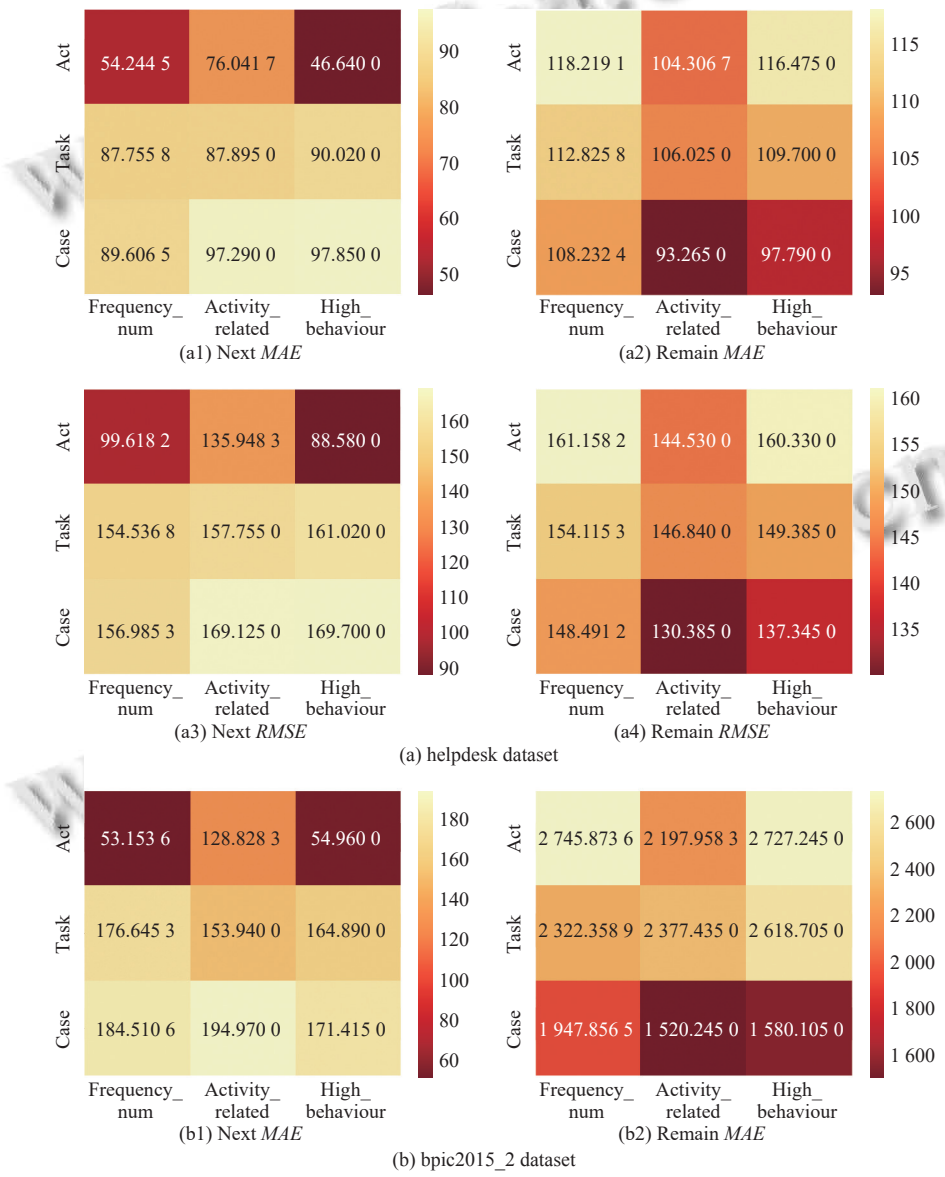


图 8 helpdesk、bpic2015_2、sepsis 活动案例间行为维度表现热力图

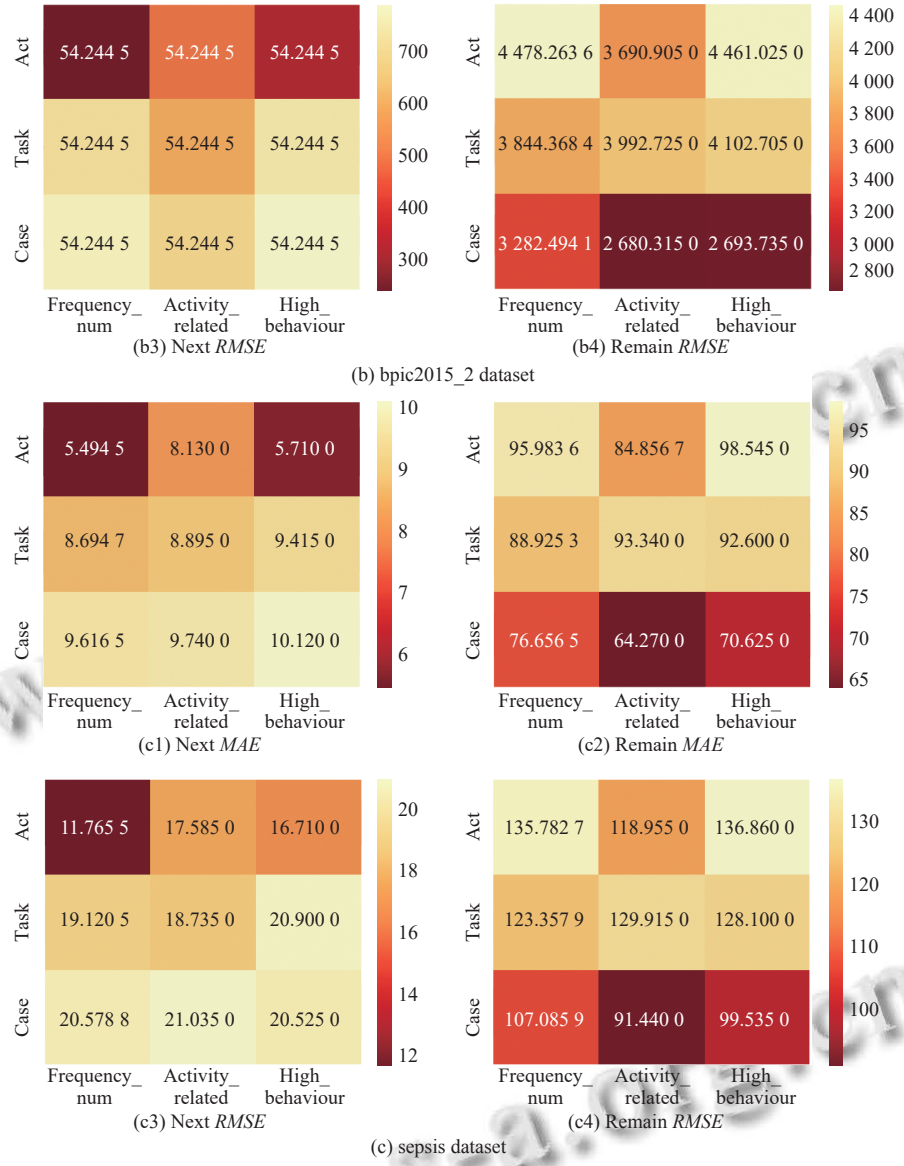


图 8 helpdesk、bpic2015_2、sepsis 活动案例间行为维度表现热力图 (续)

5 结束语

案例间行为信息的提取并编码为特征输入预测模型, 能够提高预测的准确性, 本文所做的工作是提出了一种名为 IABC 的案例间行为信息提取方法, 并将其编码为特征输入模型验证. 本文的主要贡献如下: ① 拓展了案例间行为信息的提取, 引入了活动行为视角, 提出了能够提取活动案例间行为信息的影响力分布算法与批次行为检测算法. ② 通过实验证明了活动案例间信息在业务流程监控的时间预测任务中的准确率总体优于资源案例间信息与基线方法. ③ 提供了根据预测任务不同对具体的信息维度选择的建议.

对于未来的工作, 本文只验证了单特征下案例间行为信息的有效性, 却因为特征数量的原因没有验证多特征组合下的预测准确性和结合资源视角与活动视角下综合考虑的预测准确性, 下一步计划综合考虑特征选择方法. 其次本文为回顾性研究, 前瞻性不足, 可以考虑将本文方法使用在前瞻性预测与在线预测上. 最后本文因为任务视角不同, 没有考虑下一个活动与剩余轨迹的预测, 后续研究中将考虑引入多任务方法验证活动案例间行为信息在案例轨迹预测方面的表现.

参考文献

1 Márquez-Chamorro AE, Resinas M, Ruiz-Cortés A. Predictive monitoring of business processes: A survey. IEEE

Transactions on Services Computing, 2018, 11(6): 962–977. [doi: 10.1109/TSC.2017.2772256]

2 Di Francescomarino C, Ghidini C. Predictive process monitoring. In: van der Aalst WMP, Carmona J, eds. Process Mining Handbook. Cham: Springer, 2022. 320–346.

3 Senderovich A, Di Francescomarino C, Ghidini C, *et al*. Intra and inter-case features in predictive process monitoring: A tale of two dimensions. Proceedings of the 15th International Conference on Business Process Management. Barcelona: Springer, 2017. 306–323.

4 刘聪, 张振, 郭娜, 等. 基于 Transformer 的业务流程剩余时间预测及编码方式评估方法. 山东科技大学学报 (自然科学版), 2024, 43(6): 103–112. [doi: 10.16452/j.cnki.sdkjzk.2024.06.011]

5 Folino F, Guarascio M, Pontieri L. Discovering context-aware models for predicting business process performances. Proceedings of the 2012 Confederated International Conferences on the Move to Meaningful Internet Systems: OTM 2012. Rome: Springer, 2012. 287–304.

6 Teinemaa I, Dumas M, Rosa ML, *et al*. Outcome-oriented predictive process monitoring: Review and benchmark. ACM Transactions on Knowledge Discovery from Data, 2019, 13(2): 17.

7 Bakullari B, van der Aalst WMP. High-level event mining: A framework. Proceedings of the 4th International Conference on Process Mining (ICPM). Bolzano: IEEE, 2022. 136–143.

8 Denisov V, Fahland D, van der Aalst WMP. Unbiased, fine-grained description of processes performance from event data. Proceedings of the 16th International Conference on Business Process Management. Sydney: Springer, 2018. 139–157.

9 Toosinezhad Z, Fahland D, Köroğlu Ö, *et al*. Detecting system-level behavior leading to dynamic bottlenecks. Proceedings of the 2nd International Conference on Process Mining (ICPM). Padua: IEEE, 2020. 17–24.

10 Klijn EL, Fahland D. Identifying and reducing errors in remaining time prediction due to inter-case dynamics. Proceedings of the 2nd International Conference on Process Mining (ICPM). Padua: IEEE, 2020. 25–32.

11 Senderovich A, Di Francescomarino C, Maggi FM. From knowledge-driven to data-driven inter-case feature encoding in predictive process monitoring. Information Systems, 2019, 84: 255–264. [doi: 10.1016/j.is.2019.01.007]

12 Grinvald A, Soffer P, Mokryn O. Inter-case properties and process variant considerations in time prediction: A conceptual framework. Proceedings of the 22nd International Conference on Enterprise, Business-process and Information Systems Modeling. Melbourne: Springer, 2021. 96–111.

13 Kim J, Comuzzi M, Dumas M, *et al*. Encoding resource experience for predictive process monitoring. Decision Support Systems, 2022, 153: 113669. [doi: 10.1016/j.dss.2021.113669]

14 Gunnarsson BR, Vanden Broucke S, De Weerdt J. LS-ICE: A load state intercase encoding framework for improved predictive monitoring of business processes. Information Systems, 2024, 125: 102432. [doi: 10.1016/j.is.2024.102432]

15 Bakullari B, van Thoor J, Fahland D, *et al*. The interplay between high-level problems and the process instances that give rise to them. Proceedings of the 4th International Conference on Business Process Management. Utrecht: Springer, 2023. 145–162.

16 Dubinsky Y, Soffer P, Hadar I. Detecting cross-case associations in an event log: Toward a pattern-based detection. Software and Systems Modeling, 2023, 22(6): 1755–1777. [doi: 10.1007/s10270-023-01100-w]

17 Bukhsh ZA, Saeed A, Dijkman RM. ProcessTransformer: Predictive business process monitoring with Transformer network. arXiv:2104.00721, 2021.

18 van der Aalst WMP. Object-centric process mining: Unraveling the fabric of real processes. Mathematics, 2023, 11(12): 2691. [doi: 10.3390/math11122691]

19 Pourbafrani M, van der Aalst WMP. Discovering system dynamics simulation models using process mining. IEEE Access, 2022, 10: 78527–78547. [doi: 10.1109/ACCESS.2022.3193507]

(校对责编: 王欣欣)