软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2024,35(10):4696−4709 [doi: 10.13328/j.cnki.jos.006991]

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

面向过程文本的合规性检查方法*

林雷蕾[1,3], 钱 忱[2], 闻立杰[2,3], 邱泓钧[4]

[1](首都师范大学 管理学院, 北京 100089)

[2](清华大学 软件学院, 北京 100084)

[3](工业大数据系统与应用北京市重点实验室, 北京 100084)

[4](香港科技大学 工学院, 香港 999077)

通信作者: 闻立杰, E-mail: wenlj@tsinghua.edu.cn

摘 要: 合规性检查是过程挖掘领域的重要场景之一, 其目标是判断实际运行的业务行为与理想的业务行为是否一致, 进而为业务过程管理提供决策依据. 传统的合规性检查方法存在度量指标过多、效率低等问题. 此外, 现有研究在检查过程文本与过程模型之间的合规性时严重依赖专家知识. 为此, 提出面向过程文本的合规性检查方法. 首先, 基于过程模型的执行语义生成图轨迹, 并利用词向量模型提取图轨迹中的结构特征. 同时, 引入霍夫曼树提升词向量模型的效率. 接着, 对过程文本和模型中的活动特征进行提取, 并利用孪生机制提升训练效率. 最后, 对所有特征进行融合, 并利用全连接层预测过程文本与过程模型之间的一致性得分. 实验表明, 所提方法的平均绝对误差值要比已有方法低 2 个百分点.

关键词: 过程挖掘; 孪生机制; 一致性度量; 特征表示

中图法分类号: TP311

中文引用格式: 林雷蕾, 钱忱, 闻立杰, 邱泓钧. 面向过程文本的合规性检查方法. 软件学报, 2024, 35(10): 4696–4709. http://www.jos.org.cn/1000-9825/6991.htm

英文引用格式: Lin LL, Qian C, Wen LJ, Qiu HJ. Conformance Checking Method for Process Text. Ruan Jian Xue Bao/Journal of Software, 2024, 35(10): 4696–4709 (in Chinese). http://www.jos.org.cn/1000-9825/6991.htm

Conformance Checking Method for Process Text

LIN Lei-Lei[1,3], QIAN Chen[2], WEN Li-Jie[2,3], QIU Hong-Jun[4]

[1](School of Management, Capital Normal University, Beijing 100089, China)

[2](School of Software, Tsinghua University, Beijing 100084, China)

[3](Beijing Key Laboratory of Industrial Big Data System and Application, Beijing 100084, China)

[4](School of Engineering, The Hong Kong University of Science and Technology, Hongkong 999077, China)

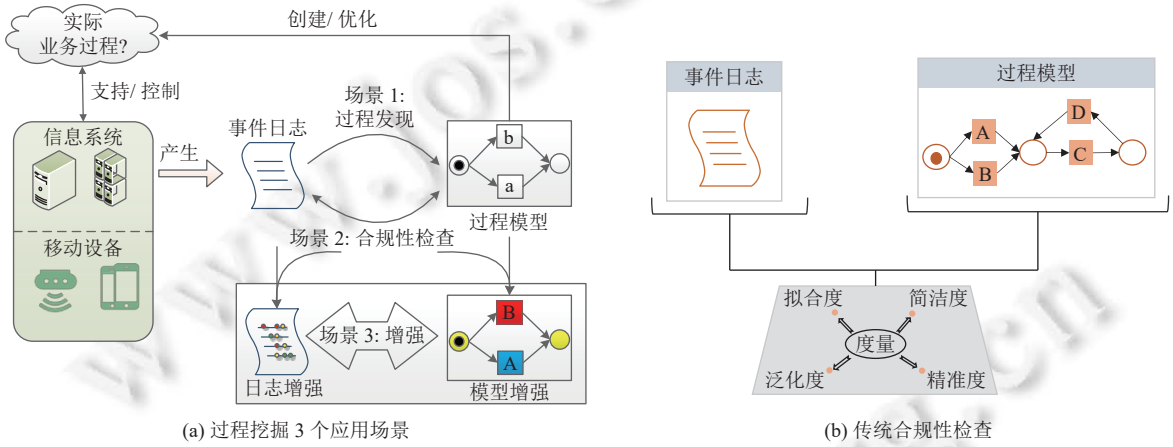
Abstract: Conformance checking is one of the important scenarios in the field of process mining, and its goal is to determine whether the actual running business behavior is consistent with the desired behavior and then provide a basis for business process management decisions. Traditional methods of conformance checking face the problems of too many metrics and low efficiency. In addition, the existing methods for checking the conformance between process text and process model rely heavily on expert-defined knowledge. Therefore, this study proposes a process text-oriented conformance checking method. Firstly, the study generates graph traces based on the execution semantics of the process model and obtains the structural features by the word vector model from graph traces. At the same time, Hoffman trees are introduced to reduce the computational effort. Then, the word vector representation of the process text and the activities is performed. The study also uses the Siamese mechanism to improve training efficiency. Finally, all the features of the text and

* 基金项目: 国家重点研发计划 (2019YFB1704003); 国家自然科学基金 (62021002); 北京市教育委员会科学研究计划 (KM202310028003)
收稿时间: 2022-08-30; 修改时间: 2023-03-01, 2023-05-25; 采用时间: 2023-06-29; jos 在线出版时间: 2023-10-18
CNKI 网络首发时间: 2023-10-19

the model are fused, and then the consistency score between the text and the model is predicted using a fully connected layer. Experiments show that the average absolute error value of the method in this study is two percentage points lower than that of existing methods.

Key words: process mining; siamese mechanism; consistency measurement; feature representation

过程挖掘 (process mining) 是业务过程管理 (business process management) 和数据挖掘 (data mining) 的交叉学科, 其目标是通过挖掘日志数据的潜在价值来赋能企业的业务过程管理[1]. 如图 1(a) 所示: 过程挖掘的 3 个场景都是从事件日志出发, 通过挖掘日志潜在价值进而创建或优化实际运行的业务过程, 最终实现提质增效、节本降耗的目的, 为企业创造更多财富. 具体讲: 1) 场景 1 是过程发现 (process discovery), 该场景解决了业务过程的自动化建模问题, 即从事件日志中准确地挖掘出业务过程模型. 为此, 国内外众多学者致力于研究出高效且准确的挖掘算法, 如遗传挖掘算法[2]、Inductive Miner 算法[3]、启发式挖掘算法[4]、Alpha 算法[5]及在 Alpha 上进行扩展的系列算法[6−8]; 2) 场景 2 是合规性检查 (conformance checking), 该场景解决的是如何度量实际业务执行的行为 (事件日志) 与理想行为 (过程模型) 之间的共性与差异[9]. 其中, 过程模型可以是通过日志中挖掘得到, 也可以是通过人工创建得到; 3) 场景 3 是过程增强 (process enhancement), 该场景解决的是如何利用领域知识来增强日志或模型. 日志增强的目标是提升数据质量, 利用已有的业务文档、业务图片、业务模型等领域数据来改进日志质量, 为后续研究提供更为准确且全面的知识[10]. 相比之下, 模型增强往往是将日志中的多维属性知识融合到业务模型中, 进而增强模型的组织维度信息、时间维度信息等[11].



(a) 过程挖掘 3 个应用场景 (b) 传统合规性检查

图 1 合规性检查在过程挖掘中的应用场景及度量内容

如图 1(b) 所示, 传统合规性检查输入是事件日志和过程模型, 输出是二者之间共性和差异. 通常用 4 个指标来度量共性和差异[12]: 1) 拟合度 (fitness), 度量的是过程模型能够重演 (replay) 多少日志行为, 即日志行为是否出现在模型中; 2) 简洁度 (simplicity), 度量过程模型在刻画同等日志行为的前提下是否是最简洁的, 即奥卡姆剃须刀原则 (Occam's razor)——"如无必要, 不应该增加多余实体"; 3) 泛化度 (generalization), 评价过程模型有多少行为不在日志中, 刻画的是模型的抽象能力; 4) 精准度 (precision), 评价过程模型的行为有多少出现在日志中, 即不允许模型存在"过多"的行为.

传统合规性检查存在的问题有: (1) 度量指标过多, 造成实际应用难以权衡. 如上文所述, 现有度量日志与模型的差异性指标有拟合度、简洁度、泛化度和精准度这 4 个指标. 这 3 个指标虽然从不同角度刻画了模型与日志的差异, 但是两两之间是相互制约的, 如泛化度和精准度, 前者希望模型具备很好的抽象能力, 而后者又限制了模型不能有过多行为. 因此, 这两个指标存在相互制约的关系, 难以权衡. (2) 度量效率低下, 限制了实时场景下大规模日志分析的应用. 以拟合度指标为例, 传统合规性检查在度量拟合度时采用的是托肯重演[13]和轨迹对齐[14]两种方式. 这两种方式都需要遍历业务模型的状态空间, 而并发与循环结构往往容易引起状态空间爆炸. 因此, 会导致传统合规性检查效率极其低下. (3) 输入限制为日志, 不利于多源多模态的场景应用. 传统合规性检查仅适用于事件

日志与过程模型之间的度量, 而实际应用中存在以文本形式记录业务流程内容的场景. 国内外众多学者也致力于研究从过程文本中发现过程模型[15,16], 并延伸到文本与模型的合规性检查[17,18]. 然而, 目前的方法严重依赖专家领域知识且误差较大.

针对上述问题, 本文提出了一种基于 TraceWalk 的网络模型, 用于解决过程模型与过程文本之间的合规性检查问题. 如图 2 所示, 本文针对的数据源是描述业务过程的过程文本 (https://cookingtutorials.com), 输出的是过程文本与已有过程模型的一致性得分 (consistency score, CS). 通过这种方式, 有效地避免了传统合规性检查存在的 3 大难点. 具体来讲: 1) 针对度量指标过多问题, 本文用差异性分值 CS 替换了原有 4 个度量指标 (拟合度、简洁度、泛化度、精准度); 2) 针对效率问题, 本文的 TraceWalk 网络模型训练结束后能快速给出分值, 不存在状态空间爆炸而得不到结果的问题. 此外, 在训练过程中本文引入了孪生机制 (siamese mechanism) 来提高训练效率; 3) 针对多源多模态问题, TraceWalk 网络模型不仅可以解决过程文本的合规性检查, 对于日志的合规性检查也可以包容, 只需在训练过程中将日志看成文本中的句子即可.

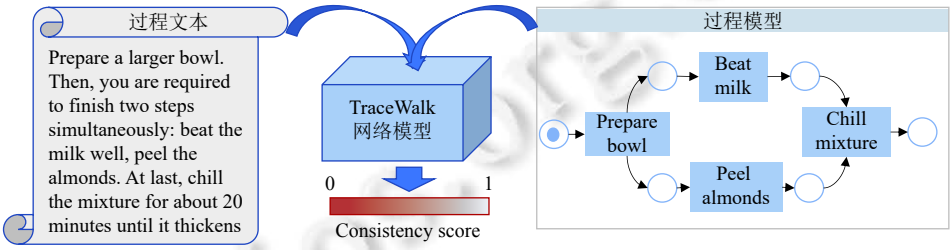


图 2 面向过程文本的合规性检查

本文第 1 节介绍合规性检查的相关工作和研究现状. 第 2 节介绍本文所需的基础知识, 包括词向量模型和随机游走等. 第 3 节介绍本文构建的基于 TraceWalk 的合规性检查模型. 第 4 节通过实验验证了本文方法的有效性. 最后对全文进行了总结.

1 合规性检查相关工作

合规性检查是过程挖掘中不可或缺的场景, 可通过度量日志与模型之间的差异, 为模型发现算法的优化提供支持. 此外, 也可以通过合规性检查来监控实际业务 (日志) 与理想业务 (业务模型) 间的偏差, 进而快速发现业务异常, 保障业务运转的连续性. 本文重点关注的是过程文本与过程模型之间的合规性检查, 融合深度学习的前沿成果, 利用特征融合等方式完成一致性度量.

Rozinat 等人[13]将轨迹放到 Petri 网模型中重放, 通过计算缺失或多余的托肯来度量日志与模型的合规性. Adriansyah[14]提出了将模型转为字母表语言 (字母代表模型中的活动名称), 然后利用轨迹和语言进行活动对齐的方式来计算合规性. 但是相对于重放, 对齐方式还多了一个寻找最优对齐的步骤, 所以会更加耗时. Burattin 等人[19]基于日志中的行为模式, 提出了在线合规性检测算法. 算法主要分为两个部分: 1) 离线部分, 将 Petir 网表达的过程模型展开 (unfolded Petri net), 从而得到模型中的可达图 (用二维矩阵表示可达图中行为先后次序); 2) 在线部分, 将实时获取到的日志数据刻画成二元关系 (活动先后次序), 最后计算二元关系与可达图之间的偏差. Peeperkorn 等人[20]利用 RNN 技术进行合规性检查. 首先, 利用模型产生模型日志和反日志 (antilog), 其中反日志就是在原有模型日志的基础上进行随机篡改得到. 然后, 通过两个日志训练得到业务过程的 RNN 模型——MRNN. 同理, 利用日志和其反日志训练出日志的 RNN 模型——LRNN. 接着, 分别将模型日志和日志数据放到 LRNN 和 MRNN 中计算出准确率和召回率, 用两个指标来度量合规性. Bauer 等人[21]为提升对齐检查的效率, 结合轨迹采样 (trace sampling) 和结果近似 (result approximation) 两种手段进行合规性检查. 但是, 由于采用了随机采样和最大近似, 其度量结果的准确率有所下降. Berti 等人[22]为了解决托肯重放方法 (token-based replay, TR) 中的托肯泛滥问题, 引入了根因分析方法对 TR 进行了优化. Felli 等人[23]利用编码 (encoding) 和解码 (decoding) 技术检查业务执行过程中数据流、

控制流等多个视角 (multi-perspective) 的偏差. Leemans 等人[24]发现当前衡量日志与模型之间的度量指标过多且难以权衡, 为此使用推土机距离 (earth mover's distance) 作为单一度量指标. 其核心思路是将模型转为接近日志形式的随机语言 (stochastic language), 接着计算语言与日志之间的推土机距离. Polyvyanyy 等人[25]利用准确率和召回率来度量日志与模型的一致性. 其中, 准确率刻画的是模型的轨迹有多少在日志中, 而召回率是日志中的轨迹有多少在模型中. 模型的轨迹由有限状态机 (deterministic finite automaton) 来表示, 接着引入拓扑熵 (topological entropy) 来计算所有轨迹中不同词出现的频率与轨迹长度的比值, 进而用模型的拓扑熵和日志的拓扑熵来刻画准确率和召回率. Sànchez-Ferreres 等人[17,18]利用过程文本和过程模型的对齐来检验文本与模型之间的合规性. 但是, 不同的对齐策略对结果影响非常大, 常用的策略有: 最佳优先搜索 (best-first searching) 和整数线性规划 (integer linear programming). 此外, 这些策略都存在两大难题: 一方面是标记和解析自然语言的成本代价非常高, 另一方面是需要特定领域的对齐规则和评估指标, 导致方法的泛化性和适应性变弱. Watanabe 等人[26]基于语法树来计算过程模型与日志轨迹之间的拟合度. 该方法首先将过程模型和日志都转为树结构, 然后基于 EM 算法 (expectation-maximization algorithm) 来计算每条轨迹与过程树之间的出现概率, 进而统计出模型与日志的拟合度. Felli 等人[27]针对日志中存在的时间不确定、活动不确定等现象, 提出了基于可满足性模理论 (satisfiability modulo theories) 的轨迹对齐方法.

2 基础知识

本文研究内容涉及 Petri 网, 事件日志和深度学习的知识, 下面就相关概念和基本知识予以介绍.

2.1 模型与日志

业务过程模型的表示符号有很多种, 常见的有因果网 C-net、Petri 网和 BPMN 等[28]. Petri 网由于其坚实的数学基础以及能很好刻画并发系统结构的优点, 在学术界得到了广泛应用. 经过多年发展, Petri 网模型包括基本标识 Petri 网、P/T 系统和着色 Petri 网[29], 本文只应用到基本标识 Petri 网.

定义 1 (**Petri 网**). 一个四元组 $N=(P, T; F, M_0)$ 代表一个 Petri 网, 其中,

(1) $P\cup T\neq\varnothing$ 且 $P\cap T=\varnothing$, 一般将 P 称为有穷库所集, T 视为有穷变迁集.

(2) $F\subseteq(P\times T)\cup(T\times P)$, 其中 F 表示流关系.

(3) $dom(F)\cup cod(F)=P\cup T$, 且 $dom(F)=\{x\in P\cup T|\exists y\in P\cup T: (x, y)\in F\}$, $cod(F)=\{y\in P\cup T|\exists x\in P\cup T: (x, y)\in F\}$.

(4) $M: P\rightarrow\{0, 1, 2,\ldots\}$, 习惯称 M 为 Petri 网 N 的一个标识, 而 M_0 代表 Petri 网的初始标识.

为了直观理解定义 1 的内容, 可以参看图 2 中的 Petri 网模型, 其刻画的是过程文本中描述的部分食谱业务内容. 具体来讲, 整个模型包含 4 个变迁 (矩形表示), 6 个库所 (圆圈表示), 10 个流关系 (箭头表示) 以及 1 个托肯 (蓝色点表示). 托肯在 Petri 网执行过程中也称之为令牌, 代表着 Petri 网的点火执行条件. 点火规则参看以下定义.

定义 2 (**可达标识集**). 令 $N=(P, T; F, M_0)$ 为 Petri 网, 则 N 的可达标识集 $R(M_0)$ 为满足以下条件的最小集合.

(1) 其中, $M_0\in R(M_0)$.

(2) 令 $M\in R(M_0)$, 且 $\exists t\in T$ 使得 $M[t\rangle M'$, 则 $M'\in R(M_0)$.

定义 3 (**前集和后集**). 假设 $N=(P, T; F, M_0)$ 代表 Petri 网, 且 $x\in P\cup T$, ${}^\bullet x=\{y|(y, x)\in F\}$, 则称 ${}^\bullet x$ 为 x 的前集; 若 $x^\bullet=\{y|(x, y)\in F\}$, 则称 $x^\bullet$ 为 x 的后集.

定义 4 (**点火规则**). 令 $N=(P, T; F, M_0)$ 为一个 Petri 网, 则有:

(1) 变迁 $t\in T$ 可点火, 表示为 $M[t\rangle$, 当且仅当 ${}^\bullet t\leq M$.

(2) 点火规则 $[\text{-}\rangle\text{-}\subseteq M\times T\times M$ 是满足下述条件的最小关系: $\forall M\in R(M_0), t\in T: M[t\rangle\Rightarrow M[t\rangle\ (M-{}^\bullet t+t^\bullet)$.

定义 2 刻画的是 Petri 网动态执行后的可达状态, 定义 3 中的前集和后集是 Petri 网的前后集合, 也是点火规则发生的前后条件. Petri 网点火后会产生变迁序列, 序列记录了 Petri 网执行后留下的足迹. 如图 1(b) 中的 Petri 网模型, 点火后可能产生的变迁序列有"AC""BCDC"等.

定义 5 (事件日志). 假设 T 为所有活动集合, 则 $\sigma \in T^*$ 是一条轨迹, $L \subseteq \mathcal{P}(T^*)$ 是一个事件日志.

定义 5 给出了事件日志的形式化定义, 本文用 T 表示事件日志的活动集合, 亦用 T 表示 Petri 网中变迁的集合, 主要是便于读者理解事件日志与过程模型的关系. 通过定义可以看出, 日志是轨迹的集合 (幂集表示), 而每条轨迹又是由有限个活动按照执行次序排列而成. 例如, 一个赔偿申请的业务日志片段 (表 1 所示) 可以表示为 L=[<A, B, C, D, E>, <A, C, B, D, F>], 其中 A 是活动 Register 的缩写, B 是 Check ticket, C 是 Examine casually, D 是 Decide, E 是 Pay compensation, F 是 Reject request. 如表 1 所示, 真实日志里面除了活动以外, 还会有每个事件的属性, 包括执行时间、执行角色及代价等. 需要注意的是, 轨迹有时候亦称为案例, 事件是活动执行的记录, 因此除了名称以外, 还拥有更多的属性. 所以, 合规性检查可以包含多维度的检查, 如控制流、时间流、数据流等不同维度[23]. 本文的合规性检查重点关注业务过程的执行顺序, 即控制流维度.

表 1 赔偿申请的业务日志片段[1]

轨迹ID	事件ID	事件属性				
		活动名称	执行时间	角色	代价	…
1	1000201	Register	03-10-2011, 09:11	Bob	10	…
	1000202	Check ticket	03-10-2011, 12:10	Sara	20	…
	1000203	Examine casually	04-10-2011, 11:10	Ellen	100	…
	1000204	Decide	06-10-2011, 13:00	Mike	20	…
	1000205	Pay compensation	15-10-2011, 10:10	Pete	35	…
2	1000301	Register	10-12-2011, 09:40	Sue	10	…
	1000302	Examine casually	10-12-2011, 14:55	Ellen	10	…
	1000303	Check ticket	13-12-2011, 13:50	Sara	10	…
	1000304	Decide	15-12-2011, 15:05	Mike	10	…
	1000305	Reject request	17-12-2011, 10:01	Wil	10	…
…	…	…	…	…	…	…

2.2 词向量模型

本文解决的是过程文本与过程模型之间的合规性检查, 因此涉及到如何对文本及模型进行向量化表示. 为此, 需介绍一下词嵌入 Word2Vec[30]的基本概念.

在深度学习的训练过程中, 所有数据都是以向量的形式进行输入. 最常见的词向量表示方法是 one-hot 和 Word2Vec 两种方式. One-hot 方式简单、易操作, 但是带来的问题是向量过于稀疏, 不利于存储和计算. 因为, one-hot 的词向量形式如 [0, 0,…, 1, 0,…, 0], 维度大小等于所有词的个数, 其中 1 表示第 i 个词, 其余都为 0. 另外, one-hot 词向量形式忽略了词之间的关联性, 无法客观地刻画词的相似性. 如, one-hot 会认为 man/woman/king/queen 是 4 个独立的词. 然而, 直觉上 4 个词是存在某种联系的. 通过 Word2Vec 的表示, 可以发现| man-woman |约等于| king-queen |, 其中|·|代表词向量之间的距离. 本质上, Word2Vec 是通过全连接的神经网络将 one-hot 编码转成低维度的连续值, 从而避免了维度灾难和词语鸿沟问题. 如图 3 左图所示, 令词汇量的大小为 V, 隐藏层的单元个数为 N, 网络模型中相连层的节点是全连接, 输入是每个词语的 one-hot 向量, 表示为 $\{x_1, x_2, \ldots, x_v\}$, 其中只有某个节点的值为 1, 其他为 0. 输出的是每个单词的预测结果 y, 因为是对 V 的词进行预测, 因此输出层大小也为 V.

由以上可知, 输入层与输出层之间的权重可以由矩阵 W 表示:

$$W = \begin{pmatrix} w_{11} & w_{12} & \cdots & w_{1n} \\ w_{21} & w_{22} & \cdots & w_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ w_{v1} & w_{v2} & \cdots & w_{vn} \end{pmatrix} \tag{1}$$

其中, W 矩阵每一行代表某个词的 N 维向量表示. 假设给定输入是某个单词 K, 其对应的 one-hot 词向量表示为第

k 个元素 x_k=1, 剩下的 $x_{k'}=0$, $k\neq k'$, 则有:

$$h=x^{\mathrm{T}}W=W_{(k,\cdot)}x_k=V_{wi}^{\mathrm{T}} \tag{2}$$

h 向量完全由 W 矩阵第 k 行计算的, 因为除了 x_k=1 外, 其余都为 0. 因此, V_{wi} 就是输入词 K 的向量表示.

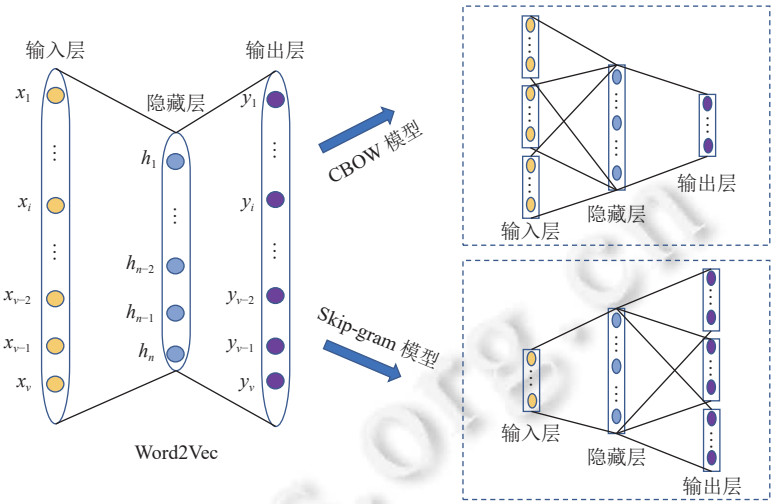


图 3 词向量表示模型

当然, 以上假设的 Word2Vec 模型是通过词的输入及分类来训练得到词向量表示. 在实际使用中, 根据不同的训练目标可以设置不同的词向量表示模型. 如图 3 右图所示, CBOW (continuous bag-of-word model) 模型通过输入中心词的上下文来预测中心词本身. 如, “I love mama”, 若选 love 为中心词, 输入的是 I 和 mama 的 one-hot 向量, 输出的是 love 向量. 在输入层, 首先要对输入的词向量进行累加平均:

$$h=\frac{1}{C}W(x_1+x_2+\ldots+x_c)=\frac{1}{C}(V_{w1}+V_{w2}+\ldots+V_{wc}) \tag{3}$$

相反, Skip-gram 模型是输入中心词, 来预测该词的上下文. de Koninck 等人采用 CBOW 模型对事件日志进行向量表示, 并在过程挖掘的聚类场景中取得不错的效果[31]. 因此, 本文采用的 Word2Vec 模型为 CBOW 模型, 默认窗口大小为 3.

3 基于 TraceWalk 的合规性检查方法

鉴于神经网络强大的表示能力和特征提取能力, 本文提出一种基于 TraceWalk 的合规性检查方法, 该方法主要包括 3 部分 (如图 4 所示): (1) 第 1 步, 首先利用 TraceWalk 的方法来刻画过程模型的结构信息. 其主要步骤是先生成图轨迹, 再通过 Word2Vec 对节点进行编码, 并利用卷积求得每个节点元素的向量; (2) 第 2 步, 对过程模型中活动信息和过程文本中的活动信息进行向量表示. 利用 Petri 网的点火规则 (定义 4) 产生过程模型的活动序列, 然后再用 Word2Vec 对活动序列和过程文本中的活动进行编码; (3) 第 3 步, 对以上 3 种特征向量采用拼接的方式进行融合, 最后计算一致性分值用以度量过程模型与过程文本之间的偏差. 需要注意一点, 在对活动序列和文本进行编码时, 分别采用了两次孪生机制, 其主要作用是提升训练的效率. 由于合规性检查涉及到实时在线场景及大规模数据场景, 因此, 效率也是需要重视的问题.

3.1 基于 TraceWalk 的模型结构特征表示

结合深度学习技术进行合规性检查是当前过程挖掘领域发展的趋势, 如文献 [20] 利用 RNN 来进行合规性检查, 但是已有方法容易忽略了模型的结构信息. 传统的图嵌入 (graph embedding)[32]方式虽然可以将图的节点转成向量, 但容易忽略节点类型和执行语义. 换句话说, 已有的图嵌入方式编辑的是相同类型的图节点, 而且节点之间的连接不代表前后执行的顺序. 然而, 过程模型中的节点至少包含两种或以上不同类型的节点, 同时节点之间存在

严谨的执行语义, 不能随便更换顺序. 如本文采用的是 Petri 网模型, 则根据定义 1 可知, 过程模型中存在库所和变迁两种类型的节点, 而且这两种类型节点是不能同类型相连接的. 具体讲, 即库所后面连接的节点必须为变迁(反之亦然). 另外, 模型的执行语义刻画了过程模型中活动执行的先后顺序. 如图 5 所示, 列举了 Petri 网 4 种基本结构: 顺序结构、选择结构、并发结构和循环结构. 其中, 顺序结构代表着活动 A 执行后, 活动 B 才能执行; 选择结构代表着活动 C 和活动 D 只能选择一个执行; 并发结构代表着活动 E 与活动 F 可以任意先后执行或同时执行; 循环结构代表活动 K 和活动 H 可以执行多次. 需要注意的是, 本文重点研究模型与文本之间的相似性, 对文献 [6] 提到的多种循环结构暂不考虑.

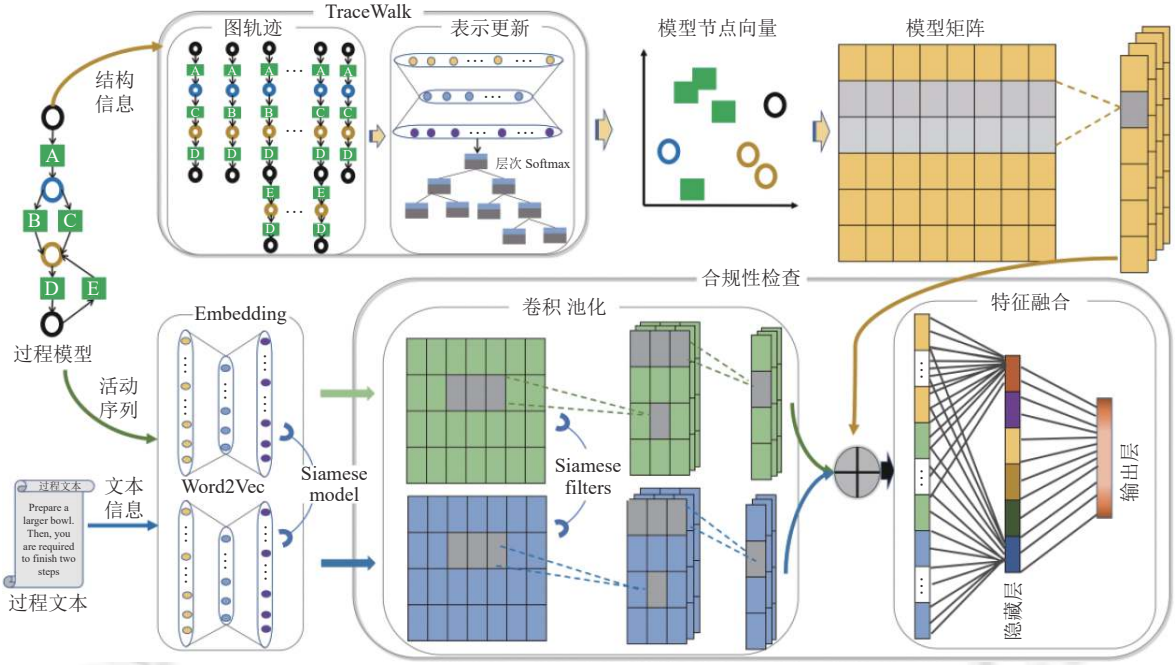


图 4 面向过程文本的合规性检查框架

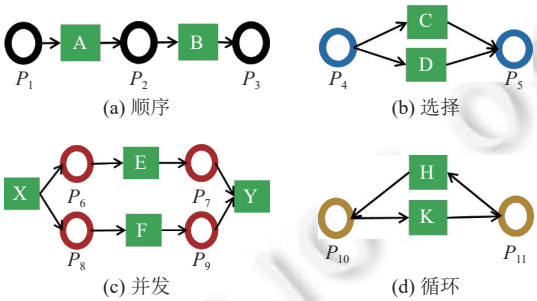


图 5 Petri 网 4 种基本结构

针对上述问题, 本文提出了基于 TraceWalk 的模型结构特征表示方法. 该方法主要包括两个部分内容: 图轨迹生成器和节点表示更新器, 其中图轨迹生成器的目标是根据过程模型的执行语义生成一条条执行路径, 然后再通过节点表示更新器来学习每个节点的向量.

如图 5 所示, 过程模型中的节点主要包括库所 (圆圈表示) 和变迁 (矩形表示) 两种, 其中变迁代表的是业务过程中的活动, 库所可以理解为活动执行需要的资源. 因此, 活动前面的资源如果存在竞争, 就会导致活动选择执行, 如果不存在资源竞争且资源充足, 则活动为并发关系. 根据 Petri 网的执行语义, 可以得到过程模型从开始到结束

的一条完整执行路径, 这条路径称之为图轨迹.

定义 6 (图轨迹). 令 N=(P, T; F, M_0) 为一个 Petri 网, $\mathcal{J}$ 为一条图轨迹, 则有 $\mathcal{J}=\tau_1,\tau_2,\ldots,\tau_n$, 其中 $\forall\tau\in P\cup T$ 且满足: $\tau_i\cap\tau_{i+1}=\varnothing$, $1\leqslant i<n$. 图轨迹的集合表示为 $\mathcal{G}=\mathcal{J}^1,\mathcal{J}^2,\ldots,\mathcal{J}^N$.

由定义 6 可得, 图轨迹是由 Petri 网中库所和变迁组成的一条路径. 由于 Petri 网是一个二分图, 因此路径中相连两个节点不能是同一类型. 图 5(b) 所示, 该选择结构包括两条图轨迹 P_4,C,P_5 和 P_4,D,P_5. 需要注意, 图轨迹虽然与日志格式很像, 但是内容不同: 图轨迹包含了模型所有节点信息, 包括库所节点. 而日志中仅记录活动信息即变迁节点. 比如, 图 5(c) 中 E 和 F 是并发关系, 则日志中会出现<E, F>和<F, E>两种先后执行顺序, 但是图 5(d) 的短循环也可以产生包含此类先后顺序的日志轨迹. 因此, 日志中活动信息无法准确地反应模型结构信息. 但是, 图轨迹通过引入库所节点, 就可以利用变迁前后库所信息来识别两种结构的不同之处. 换句话, 图轨迹通过引入库所节点, 能更加准确刻画过程模型的结构信息.

随机游走[30]能够从有向图中抽取出图序列, 因此, 课题组借鉴 DeepWalk 的模块来生成图轨迹. 核心思路是给定一个起始节点 τ, 以及序列长度 t, 在执行语义的基础上利用深度优先的方式遍历节点. 由于流程模型存在循环结构, 因此默认的序列长度是 max=1000. 换句话说, 图轨迹产生过程有两个终止条件: ① 遍历到流程的结束节点; ② 图轨迹长度达到上限 max. 对于图轨迹的集合 $\mathcal{G}=\mathcal{J}^1,\mathcal{J}^2,\ldots,\mathcal{J}^N$ 可以看成是自然语言中的段落, 其中每条轨迹 $\mathcal{J}=\tau_1,\tau_2,\ldots,\tau_n$ 是一个句子. 在此基础上, 可以采用词向量模型来训练图轨迹的节点表示, 所以 TraceWalk 的目标是寻找图轨迹中出现节点 τ_t 后, 出现后续节点 τ_{t+j} 的最大概率:

$$\frac{1}{N}\sum_{i=1}^{n}\left(\frac{1}{n}\sum\nolimits_{t=1}^{n}\sum_{-c\leqslant j\leqslant c,j\neq 0}\log p(\tau_{t+j}|\tau_t)\right) \tag{4}$$

其中, c 表示训练内容的大小, 其值越大, 训练的准确率越高, 但也带来了耗时问题. 流程模型中存在循环和并发结构, 图轨迹内容会非常庞大. 因此, 本文采用 Softmax 函数来定义公式 (4) 中的 $p(\tau_{t+j}|\tau_t)$, 并利用 hierarchical Softmax 来降低训练时间. 具体的 Softmax 内容是:

$$p(\tau_O|\tau_I)=\frac{\exp\left(v'^{\mathrm{T}}_{\tau_O}v_{\tau_I}\right)}{\sum\limits_{v=1}^{V}\exp\left(v'^{\mathrm{T}}_{\tau_v}v_{\tau_I}\right)} \tag{5}$$

其中, v'_w 和 v_w 分别代表节点 w 的输入和输出向量, V 表示图轨迹中的节点数量. 接着, 引入霍夫曼树 (Huffman binary tree) 来搭建层次 Softmax. 对于任意一个节点 v_j, 其向量表示为 $\Phi(v_j)\in\mathbb{R}^d$. 如果存在一条路径 $<n_0,n_1,\ldots,n_{\log|V|}>$ 到达节点 w_k, 其中 n_0 =root, $n_{\log|V|}=w_k$, 则:

$$p(w_k|\Phi(w_I))=\prod_{l=1}^{\lceil\log|V|\rceil}\frac{1}{1+\exp(-[[n_l]]v'^{\mathrm{T}}_{n_l}v_{w_I})} \tag{6}$$

其中, $[[n_l]]$ 取值为 1 或者−1. 如果取值为 1, 则表明节点 n_l 是节点 n_{l-1} 的左孩子. 如果取值为−1, 则为右孩子. 通过霍夫曼树的优化, 可以将计算复杂度从 O(|V|) 降到 O(log|V|).

经过图轨迹的生成和节点向量的表示后, 将所有节点向量整理得到模型矩阵, 其中每一行代表一个节点. 紧接着, 为了强化节点之间的次序信息, 借鉴 TextCNN 抽取 N-gram 思想, 利用卷积从上往下抽取出节点的次序特征. 卷积核尺度 N 分别取值为 N = 2, N = 3 及 N = 5, 步长为 1. 其中, 数值 N 代表卷积核获取到的节点数量, 2 代表的是获取到相邻两个节点的前后次序, 不同尺度可以捕捉到不同次序特征. 需要注意一点, N 的值不宜过大, 一般取值范围是 [2, 5]. 最后, 将卷积后的所有向量拼接在一起, 得到基于 TraceWalk 的模型结构特征 $\mathcal{S}$.

3.2 活动序列与文本特征表示

将模型结构进行向量化表示后, 接下来需要将模型中的活动信息以及过程文本信息也表示为向量. 前者是捕获模型结构信息, 后者是捕获模型中的活动信息和业务过程描述文本中的词语信息. 因此, 本文使用预训练的 Word2Vec 模型 (https://code.google.com/archive/p/word2vec) 来抽取后者的词语信息.

其中, 活动信息是将模型点火产生的活动序列作为输入, 词语信息是将流程描述文本作为输入. 以图 2 为例,

过程模型点火产生的活动序列为<prepare bowl, beat milk, peel almonds, chill mixture>, <prepare bowl, peel almonds, beat milk, chill mixture>. 过程文本中的所有词都会被输入到网络中, 通过 Word2Vec 方式拿到每个词的向量. 可以看到, 活动序列中出现的词都是从过程文本中提取出来得到的. 换而言之, 词向量模型在训练过程中关注的内容是一样的. 因此, 课题组采用孪生机制[33]来训练词向量模型, 既保证了特征的准确性, 又提高了模型的效率. 同样, 在卷积过程中, 也引入了孪生机制来共享卷积核. 孪生机制的本质就是共享一个能量函数 (energy function), 进而保证模型或卷积核的权值是共享的.

令 M, N 为两个词向量模型, $F^a =< f_1^a, f_2^a, \ldots, f_m^a >$ 和 $F^b =< f_1^b, f_2^b, \ldots, f_n^b >$ 是两个卷积核集合. 则:

$$\begin{cases} M \equiv N \\ m \equiv n \\ f_i^a \equiv f_i^b, \forall i \in [1,2,\ldots,m] \end{cases} \tag{7}$$

公式 (7) 表明了词向量模型和卷积核的结构是相等且权值共享的. 同时, 在卷积过程中引入偏差和非线性函数, 使得整个最终的活动序列特征 $\mathcal{R}$ 及文本特征 $\mathcal{P}$ 能够尽可能包含每个词的上下文信息. 卷积过程为:

$$v = \left[\left[\begin{array}{c} \sigma(w_i x_1^h + b_i) \\ \sigma(w_i x_2^{h+1} + b_i) \\ \vdots \\ \sigma(w_i x_{n-h+1}^n + b_i) \end{array}\right]\right] \tag{8}$$

其中, x_1^n 表示向量 $x_1, x_2, \ldots, x_n$ 的级联操作, b_i 是偏差, $\sigma(\cdot)$ 表示非线性操作, v 是通过一次卷积后得到新的特征向量. 此外, 在卷积过程中为了突出关键词重要性 (即: 过程模型中的活动), 采用的是最大池化 (max-pooling) 机制.

3.3 特征融合

前面通过 TraceWalk 获取到的结构特征 $\mathcal{S}$, 并利用孪生机制获取到的活动序列特征 $\mathcal{R}$ 及文本特征 $\mathcal{P}$. 接着, 需要考虑如何进行特征融合. 常见的融合方式有两种: 早融合 (early fusion) 和晚融合 (late fusion). 晚融合是针对不同模型的预测结果进行融合, 适合多模型结构. 为此, 本文选取的策略是早融合中的系列特征融合方法 (concat). 这种策略是将两个特征直接拼接在一起, 通过全连接层的大量训练来预测出最优结果.

$$\begin{cases} \mathcal{V} = \mathcal{S} \oplus \mathcal{R} \oplus \mathcal{P} \\ O_i = Softmax(W_2 \cdot (W_1 \cdot \mathcal{V} + b_1) + b_2) \end{cases} \tag{9}$$

其中, $\mathcal{V}$ 为 3 个特征融合后的向量, O_i 表示经过 3 层全连接后输出的结果.

令 x_i 表示第 i 次训练样本, y_i 表示样本的真实值, e_i 表示样本的预测值, 则整个深度模型的最小损失函数为:

$$J(\theta, \gamma) = \frac{1}{M}\left(\sum_{i:e_i \leqslant y_i} \gamma |e_i - y_i| + \sum_{i:e_i > y_i} (1-\gamma)|e_i - y_i|\right) \tag{10}$$

其中, M 为样本数量, $\gamma \in [0.0, 1.0]$ 为调整因子, 是模型的一个超参数. 本文在训练模型的过程中以一个批次 (batch) 反向传播来更新深度模型的参数, 而一批次中就包含预测值与真实值差距大的样本, 也包含了预测值和真实值差距小的样本. 公式 (10) 的目标是当预测值 e_i 与真实值 y_i 差距较小时, 模型参数调整的幅度小. 当预测值与真实值差距较大时, 模型中参数的调整幅度也随之变大. 因此, γ 的取值大小会影响到模型参数调整的幅度. 当 γ=0.5 时, 公式 (10) 就相当于常用度量指标平均绝对误差 *MAE* (mean absolute error). 本文默认 γ 的取值为 0.5.

4 实验分析

4.1 实验数据

过程文本与过程模型的合规性检查是过程挖掘领域近几年研究热点, 目前尚未看到公开发表的数据集. Sànchez-Ferreres 等人是通过人工合成方式对已有文本数据进行标注[18], 但是该数据集是为了从文本中发现过程模型, 打标的内容与结果跟本文不相符. 因此, 采用文献 [34] 提出的 Goun 方法进行过程文本生成, 然后利用行为

轮廓 (behavior profile)[35]的方式对数据集进行相似度打标. 需要注意的是, 文献 [36,37] 在行为轮廓基础上更加全面地考虑了异步并发及复杂变迁关系的一致性检测, 但是本文采用的数据集极少包含以上结构, 因此在不影响整体效果前提下采用了效率更好的行为轮廓方法进行打标. 具体打标步骤如图 6 所示: 首先, 采用 Goun 方法将两个过程模型 G_i、G_j 分别生成文本描述 T_i、T_j. 然后, 利用行为轮廓方法计算出两个过程模型的真实相似度 $BP(G_i, G_j)$ 作为标准答案. 最后, 将业务过程模型 G_i、文本描述 T_j 和标准答案 $BP(G_i, G_j)$ 作为一个样本, 存入训练数据集.

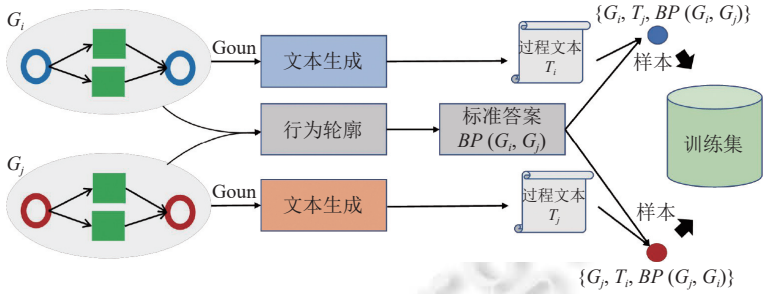


图 6 训练集生成过程

合规性检测需要模型与文本配对数据集, Goun 方法能够解决这一难题. 然而, Goun 方法输入的是过程模型. 为此, 本文选取了 4 组过程模型的数据集用于训练 (如表 2 所示).

表 2 实验数据集

分组	类型	模型数量	平均活动数量	结构化占比 (%)
RGG	①	2 284	23.0	47
SPR	②	394	7.7	100
IPN	②	1 222	50.4	76
ABM	③	602	37.5	42

注: ①: 课题组人工生成数据; ②: 工业界真实数据; ③: 已发表论文的数据

(1) 随机生成过程模型 (RGG). BeehiveZ (https://github.com/lcynju/BeehiveZ-NJU) 是国内开发的一款专门用于过程挖掘的开源平台, 该平台已在多个高校及工业界适用. 为此, 本文第 1 组过程模型数据是采用 BeehiveZ 生成.

(2) 过程仓库 (SPR). SAP 公司 (https://www.sap.com) 是全球著名的 ERP 供应商, 其致力于打造高效的企业管理流程. 因此, 从该公司的模型仓库中收集了第 2 组模型数据.

(3) 工业运营数据 (IPN). DG、TC 和 IBM 都提供过工业运营中的 Petri 网模型[34]. 为此, 选用这些数据作为第 3 组模型数据.

(4) 学界过程模型 (ABM). 最后一组数据来自于文献 [18]、文献 [34] 和 BAI (https://bpmai.org) 提供的过程模型.

需要注意, 以上数据内容主要由 Petri 网 (第 1 组和第 3 组) 和 BPMN (第 2 组和第 4 组) 两种模型格式. 对于 BPMN 格式, 可用过程挖掘的公认开源平台 ProM 转为 Petri 网.

4.2 实现细节及基准模型

本文涉及内容采用的是 TensorFlow 框架进行实现, 其中词编码采用的大小为 100, 卷积核的个数为 128, 在特征融合中每层是 128 个单元. 此外, 单元的激活函数采用 Sigmoid, 每个批量后的优化函数是 Adam. 训练的轮次是 1 万, 最小批量是 128, 其中学习率为 0.000 2. 为了能找到最优解, 在训练 7 000 轮次后会调整为 0.000 1. 受文献 [38] 的启发, 所有参数的初始值设定为 $[-\sqrt{6/(r+c)}, \sqrt{6/(r+c)}]$, 其中 r 和 c 分别代表矩阵的长和宽. 在配备 3 个 GeForceGTX-1080-Ti GPU 的计算机上, 训练阶段平均耗时 30 min.

本文选择以下 4 个方法进行比较.

(1) FCCM. 由 Han 等人[39]提出的第 1 个针对过程文本与过程图之间的一致性检查方法. 该方法的策略是将文

本中的任务出现次序与过程图中的活动依次对齐, 计算二者一致性.

(2) MACO. 利用丢失的活动 (missing activities) 和冲突的次序关系 (conflicting orders) 来计算文本与模型之间的差异性[40].

(3) ILP. Ferreres 等人[17]将文本和模型的对齐过程转为数学优化问题, 进而求解差异性.

(4) PIIMC. 文献 [18] 制定了 6 种对齐规则, 然后通过 6 种规则抽取出指定的特征内容, 最后通过计算特征间的距离来表示二者差异性.

4.3 实验结果与分析

为了评估本文方法在合规性检查上的表现, 实验结果预回答以下两个问题.

• RQ1: TraceWalk 在过程文本与过程模型的合规性检查上是否获得更好的效果?

• RQ2: TraceWalk 网络模型中核心模块是否可靠?

RQ1: TraceWalk 在过程文本与过程模型的合规性检查上是否获得更好的效果?

首先, 验证本文方法相对于已有合规性检查方法是否有提升. 为此, 将 TraceWalk 方法与 FCCM、MACO、ILP 和 PIIMC 这 4 种方法进行对比实验. 同时, 为了更为全面地展示不同方法的表现能力, 将实验对象分为两类任务: 第 1 种, 活动层级 (A-task), 只关注模型活动与文本之间的一致性; 第 2 种, 模型层级 (M-task), 这层级不仅考虑活动还要把活动的次序关系也考虑进来, 实验的结果见表 3.

表 3 TraceWalk 与 4 种方法结果比较

方法	A-task				M-task			
	RGG	SPR	IPN	ABM	RGG	SPR	IPN	ABM
FCCM	0.102	0.202	0.115	0.188	0.347	0.303	0.269	0.389
MACO	0.076	0.163	0.069	0.126	0.295	0.284	0.215	0.333
ILP	0.062	0.126	**0.043**	0.068	0.250	0.269	0.164	0.295
PIIMC	**0.053**	0.117	**0.043**	0.059	0.241	0.266	0.156	0.290
TraceWalk	0.058	**0.112**	0.058	**0.056**	**0.109**	**0.134**	**0.064**	**0.067**

在活动层级上, 可以看到 FCCM 和 MACO 两个方法的绝对误差值都比较高, 所以合规性检查的结果不如 ILP、PIIMC 和 TraceWalk. 此外, ILP 和 PIIMC 两个方法在 IPN 的数据集表现都超过了 TraceWalk 方法, 前两者的平均绝对误差为 0.043, 而 TraceWalk 的误差为 0.058. 但是, 在 SPR 和 ABM 数据集上 TraceWalk 的平均绝对误差又是最小的. 因此, 可以得出在活动层级的识别及对齐方面, TraceWalk 与 PIIMC 两种方法表现较好. 其中, PIIMC 方法在合规检查过程中制定了大量复杂的规则, 严重依赖于专家的领域知识, 而 TraceWalk 方法通过深度学习方式, 自动获取到文本与模型中的活动对齐方式.

在模型层级上, 可发现 ILP 与 PIIMC 的方法优于 FCCM 和 MACO. 但是, TraceWalk 在模型层级的表现更好. 相对于 FCCM 和 MACO, TraceWalk 的平均绝对误差要低 2 个百分点; 相对于 ILP 和 PIIMC, TraceWalk 的 MAE 得分至少低 1 个百分点, 在 ABM 数据集上, 更是低了 2 个百分点. 所以, 可以看出现有方法在合规性检查中对于活动间次序关系的处理存在明显不足.

综上, 本文方法 TraceWalk 在过程文本和过程模型的合规性检查相对于现有的方法更具有实用价值.

以上 5 种方法都是输出文本与过程模型之间的一致性得分, 为了衡量不同方法之间的准确率, 采用平均绝对误差 (*MAE*):

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i' - y_i| \tag{11}$$

其中, n 代表的是所有文本与模型对的数量, y_i' 为 5 种方法识别的一致性得分, y_i 为真实得分.

RQ2: TraceWalk 网络模型中核心模块是否可靠?

其次, 验证本文方法每个模块的可靠性. 为此, 我们将对 TraceWalk 进行消融实验 (ablation experiment). 该方

法中核心模块是利用执行语义将模型展开成图轨迹, 再对图轨迹进行向量表示, 获得活动之间的次序关系. 对比之下, 将这个核心模块去掉, 形成“TraceWalk-∅”, 将这个模块替换成 DeepWalk, 直接在模型上游走, 形成“TraceWalk-D”. 通过 3 种方式的对比, 来验证本文提出的网络模型可靠性, 实验结果见图 7.

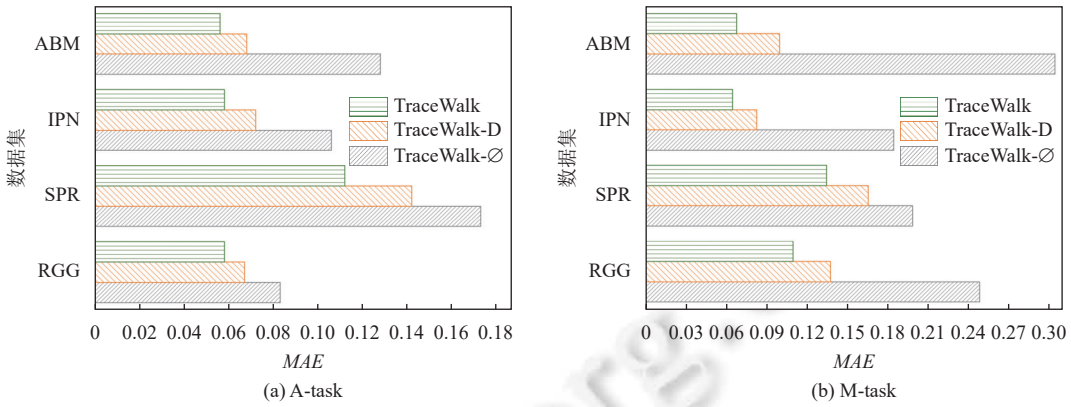


图 7 TraceWalk 模块的在两个类型任务中得分

如图 7 所示, 无论是在 A-task (图 7(a)) 还是 M-task (图 7(b)) 两种类型的任务上, TraceWalk-∅ 的平均绝对误差值 *MAE* 都是最高的, 说明仅靠简单的 Word2Vec 很难实现文本与模型之间的合规性检查. 本质上, 还是因为 TraceWalk-∅ 忽略了活动之间的次序关系. TraceWalk-D 是在 Word2Vec 的基础上对过程模型进行随机游走, 这种方式虽然能获取到一定的活动次序关系, 但存在关系不全且错误的情况. 相反, 采用了 TraceWalk 的深度模型, 是基于图轨迹来获取活动次序关系, 而图轨迹是根据过程模型执行语义展开的, 通过这种方式来捕获到的活动次序关系比采用 DeepWalk 直接在过程模型上捕捉的次序关系要更加准确, 因此 TraceWalk 的 *MAE* 得分是最低的, 进而表明本文提出的深度学习模型中, 每个模块都是合理且可靠的.

综合前面的对比实验和消融实验分析, 可得出本文提出的合规性检查方法在不同层级的任务上, 性能都优于已有方法. 此外, 本文方法的核心模块能较好地捕捉到活动次序关系, 保证了方法的有效性.

5 总 结

合规性检查是过程挖掘领域 3 大应用场景之一, 其主要检查数据与过程模型之间的一致性. 针对现有合规性检查方法存在指标过多、状态空间爆炸等问题, 本文提出了一种基于 TraceWalk 的合规性检查方法, 该方法能够较好解决过程文本与过程模型之间的一致性检查问题. 本文方法主要采用图轨迹来捕获活动次序, 再利用 Word2Vec 编码表示文本知识, 最后通过特征融合手段来对文本及过程模型进行一致性检查. 实验表明, 已有方法虽然能检查到文本中活动与模型中活动的一致性, 但很难获取到活动之间次序关系. 对比之下, 本文方法在多个数据集上, 整体表现优异. 在不依赖任何专家规则的基础上, 平均绝对值误差相对已有方法低 2 个百分点. 下一步, 将重点讨论复杂模型结构的文本生成及合规性检查, 并研究如何准确定位到差异性内容且给出合理描述.

References:

[1] van der Aalst WMP. Process Mining: Discovery, Conformance and Enhancement of Business Processes. Berlin: Springer, 2014. [doi: 10.1007/978-3-642-19345-3]

[2] de Medeiros AKA, Weijters AJMM, van der Aalst WMP. Genetic process mining: An experimental evaluation. Data Mining & Knowledge Discovery, 2007, 14(2): 245–304. [doi: 10.1007/s10618-006-0061-7]

[3] Leemans SJJ, Fahland D, van der Aalst WMP. Discovering block-structured process models from event logs—A constructive approach. In: Proc. of the 34th Int'l Conf. on Application and Theory of Petri Nets and Concurrency. Milan: Springer, 2013. 311–329. [doi: 10.1007/978-3-642-38697-8_17]

[4] Weijters AJMM, van der Aalst WMP, de Medeiros AKA. Process mining with the heuristics miner algorithm. Eindhoven: Technische

Universiteit Eindhoven, 2006.

[5] van der Aalst W, Weijters T, Maruster L. Workflow mining: Discovering process models from event logs. IEEE Trans. on Knowledge & Data Engineering, 2004, 16(9): 1128–1142. [doi: 10.1109/TKDE.2004.47]

[6] Lin LL, Zhou H, Dai F, Zhu R, Li T. Approach to mining length-two loops from the log without "aba" pattern. Ruan Jian Xue Bao/Journal of Software, 2018, 29(11): 3278–3294 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/5478.htm [doi: 10.13328/j.cnki.jos.005478]

[7] Guo QL, Wen LJ, Wang JM, Yan ZQ, Yu PS. Mining invisible tasks in non-free-choice constructs. In: Proc. of the 13th Int'l Conf. on Business Process Management. Innsbruck: Springer, 2015. 109–125. [doi: 10.1007/978-3-319-23063-4_7]

[8] Lekić J, Milićev D. Discovering block-structured parallel process models from causally complete event logs. Journal of Electrical Engineering, 2016, 67(2): 111–123. [doi: 10.1515/jee-2016-0016]

[9] Leemans SJJ, Fahland D, van der Aalst WMP. Scalable process discovery and conformance checking. Software and Systems Modeling, 2018, 17(2): 599–631. [doi: 10.1007/s10270-016-0545-x]

[10] Wang JM, Song SX, Lin XM, Zhu XC, Pei J. Cleaning structured event logs: A graph repair approach. In: Proc. of the 31st IEEE Int'l Conf. on Data Engineering. Seoul: IEEE, 2015. 30–41. [doi: 10.1109/ICDE.2015.7113270]

[11] Okoye K, Tawil ARH, Naeem U, Lamine E. Semantic process mining towards discovery and enhancement of learning model analysis. In: Proc. of the 17th IEEE Int'l Conf. on High Performance Computing and Communications, the 7th IEEE Int'l Symp. on Cyberspace Safety and Security, and the 12th IEEE Int'l Conf. on Embedded Software and Systems. New York: IEEE, 2015. 363–370 [doi: 10.1109/HPCC-CSS-ICESS.2015.164]

[12] Buijs JCAM, van Dongen BF, van der Aalst WMP. On the role of fitness, precision, generalization and simplicity in process discovery. In: Proc. of the 2012 Confederated Int'l Conf. on the Move to Meaningful Internet Systems. Rome: Springer, 2012. 305–322. [doi: 10.1007/978-3-642-33606-5_19]

[13] Rozinat A, van der Aalst WMP. Conformance checking of processes based on monitoring real behavior. Information Systems, 2008, 33(1): 64–95. [doi: 10.1016/j.is.2007.07.001]

[14] Adriansyah A. Aligning observed and modeled behavior [Ph.D. Thesis]. Eindhoven: Technische Universiteit Eindhoven, 2014. [doi: 10.6100/IR770080]

[15] Qian C, Wen LJ, Kumar A, Lin LL, Lin L, Zong Z, Li SA, Wang JM. An approach for process model extraction by multi-grained text classification. In: Proc. of the 32nd Int'l Conf. on Advanced Information Systems Engineering. Grenoble: Springer, 2020. 268–282. [doi: 10.1007/978-3-030-49435-3_17]

[16] van der Aa H, Di Ciccio C, Leopold H, Reijers HA. Extracting declarative process models from natural language. In: Proc. of the 31st Int'l Conf. on Advanced Information Systems Engineering. Rome: Springer, 2019. 365–382. [doi: 10.1007/978-3-030-21290-2_23]

[17] Sànchez-Ferreres J, Carmona J, Padró L. Aligning textual and graphical descriptions of processes through ILP techniques. In: Proc. of the 29th Int'l Conf. on Advanced Information Systems Engineering. Essen: Springer, 2017. 413–427. [doi: 10.1007/978-3-319-59536-8_26]

[18] Sànchez-Ferreres J, van der Aa H, Carmona J, Padró L. Aligning textual and model-based process descriptions. Data & Knowledge Engineering, 2018, 118: 25–40. [doi: 10.1016/j.datak.2018.09.001]

[19] Burattin A, van Zelst SJ, Armas-Cervantes A, van Dongen BF, Carmona J. Online conformance checking using behavioural patterns. In: Proc. of the 16th Int'l Conf. on Business Process Management. Sydney: Springer, 2018. 250–267. [doi: 10.1007/978-3-319-98648-7_15]

[20] Peeperkorn J, Vanden Broucke S, De Weerdt J. Supervised conformance checking using recurrent neural network classifiers. In: Proc. of the 2021 Int'l Conf. on Process Mining Workshops. Padua: Springer, 2021. 175–187. [doi: 10.1007/978-3-030-72693-5_14]

[21] Bauer M, van der Aa H, Weidlich M. Sampling and approximation techniques for efficient process conformance checking. Information Systems, 2022, 104: 101666. [doi: 10.1016/j.is.2020.101666]

[22] Berti A, van der Aalst WMP. A novel token-based replay technique to speed up conformance checking and process enhancement. In: Koutny M, Kordon F, Pomello L, eds. Trans. on Petri Nets and Other Models of Concurrency XV. Berlin: Springer, 2021. 1–26. [doi: 10.1007/978-3-662-63079-2_1]

[23] Felli P, Gianola A, Montali M, Rivkin A, Winkler S. CoCoMoT: Conformance checking of multi-perspective processes via SMT. In: Proc. of the 19th Int'l Conf. on Business Process Management. Rome: Springer, 2021. 217–234. [doi: 10.1007/978-3-030-85469-0_15]

[24] Leemans SJJ, van der Aalst WMP, Brockhoff T, Polyvyanyy A. Stochastic process mining: Earth movers' stochastic conformance. Information Systems, 2021, 102: 101724. [doi: 10.1016/j.is.2021.101724]

[25] Polyvyanyy A, Kalenkova A. Conformance checking of partially matching processes: An entropy-based approach. Information Systems, 2022, 106: 101720. [doi: 10.1016/j.is.2021.101720]

[26] Watanabe A, Takahashi Y, Ikeuchi H, Matsuda K. Grammar-based process model representation for probabilistic conformance checking.

In: Proc. of the 4th Int'l Conf. on Process Mining. Bolzano: IEEE, 2022. 88–95.

[27] Felli P, Gianola A, Montali M, Rivkin A, Winkler S. Conformance checking with uncertainty via SMT. In: Proc. of the 20th Int'l Conf. on Business Process Management. Münster: Springer, 2022. 199–216. [doi: 10.1007/978-3-031-16103-2_15]

[28] Dumas M, La Rosa M, Mendling J, Reijers HA. Fundamentals of Business Process Management. 2nd ed., Berlin: Springer, 2018. [doi: 10.1007/978-3-662-56509-4]

[29] Gehlot V. From Petri NETS to colored petri NETS: A tutorial introduction to NETS based formalism for modeling and simulation. In: Proc. of the 2019 Winter Simulation Conf. National Harbor: IEEE, 2019. 1519–1533. [doi: 10.1109/WSC40007.2019.9004691]

[30] Mikolov T, Chen K, Corrado G, Dean J. Efficient estimation of word representations in vector space. arXiv:1301.3781, 2013.

[31] de Koninck P, vanden Broucke S, De Weerdt J. Act2vec, trace2vec, log2vec, and model2vec: Representation learning for business processes. In: Proc. of the 16th Int'l Conf. on Business Process Management. Sydney: Springer, 2018. 305–321. [doi: 10.1007/978-3-319-98648-7_18]

[32] Perozzi B, Al-Rfou R, Skiena S. DeepWalk: Online learning of social representations. In: Proc. of the 20th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining, New York: ACM, 2014. 701–710. [doi: 10.1145/2623330.2623732]

[33] Mueller J, Thyagarajan A. Siamese recurrent architectures for learning sentence similarity. Proc. of the 2016 AAAI Conf. on Artificial Intelligence, 2016, 30(1): 2786–2792. [doi: 10.1609/aaai.v30i1.10350]

[34] Qian C, Wen LJ, Wang JM, Kumar A, Li HR. Structural descriptions of process models based on goal-oriented unfolding. In: Proc. of the 29th Int'l Conf. on Advanced Information Systems Engineering (CAISE). Essen: Springer, 2017. 397–412. [doi: 10.1007/978-3-319-59536-8_25]

[35] Weidlich M, Mendling J, Weske M. Efficient consistency measurement based on behavioral profiles of process models. IEEE Trans. on Software Engineering, 2011, 37(3): 410–429. [doi: 10.1109/TSE.2010.96]

[36] Wang MM, Ding ZJ, Liu GJ, Jiang CJ, Zhou MC. Measurement and computation of profile similarity of workflow nets based on behavioral relation matrix. IEEE Trans. on Systems, Man, and Cybernetics: Systems, 2020, 50(10): 3628–3645. [doi: 10.1109/TSMC.2018.2852652]

[37] Zhao F, Xiang DM, Liu GJ, Jiang CJ. A new method for measuring the behavioral consistency degree of WF-net systems. IEEE Trans. on Computational Social Systems, 2022, 9(2): 480–493. [doi: 10.1109/TCSS.2021.3099475]

[38] Bengio Y, Glorot X. Understanding the difficulty of training deep feed forward neural networks. In: Proc. of the 13th Int'l Conf. on Artificial Intelligence and Statistics. Sardinia: JMLR, 2010. 249–256.

[39] Van der Aa H, Leopold H, Reijers HA. Detecting inconsistencies between process models and textual descriptions. In: Proc. of the 13th Int'l Conf. on Business Process Management. Innsbruck: Springer, 2016. 90–105. [doi: 10.1007/978-3-319-23063-4_6]

[40] Van der Aa H, Leopold H, Reijers HA. Comparing textual descriptions to process models—The automatic detection of inconsistencies. Information Systems, 2017, 64: 447–460. [doi: 10.1016/j.is.2016.07.010]

附中文参考文献:

[6] 林雷蕾, 周华, 代飞, 朱锐, 李彤. 一种从无"aba"模式的日志中挖掘2度循环的方法. 软件学报, 2018, 29(11): 3278–3294. http://www.jos.org.cn/1000-9825/5478.htm [doi: 10.13328/j.cnki.jos.005478]

林雷蕾(1989－), 男, 博士, 讲师, 主要研究领域为流程管理, 软件工程, 大数据分析, 服务计算.

闻立杰(1977－), 男, 博士, 副教授, 博士生导师, CCF 专业会员, 主要研究领域为过程数据管理与挖掘, 交互式大数据分析, 自然语言处理.

钱忱(1994－), 男, 博士, 主要研究领域为自然语言处理, 过程挖掘.

邱泓钧(1993－), 男, 博士生, 主要研究领域为软件工程, 程序分析, 形式化方法.