基于图常量条件函数依赖的图修复规则发现

李 杰[1,2,3],曹建军[2,3],王保卫[1],庄 园[1,2,3]
(1. 南京信息工程大学 计算机学院 网络空间安全学院,江苏 南京 210044;
2. 国防科技大学 第六十三研究所,江苏 南京 210007;
3. 国防科技大学 大数据与决策实验室,湖南 长沙 410073)

摘 要:数据一致性是数据质量管理的一个重要内容。为了提升图数据一致性,大量关系型数据库中的数据依赖理论被引入到图数据库,包括图函数依赖、图关联规则等。图修复规则是最新提出的一种针对图数据的数据依赖规则,具有强大的修复能力,但目前尚无有效的挖掘算法。为了自动生成图修复规则并提高图数据修复的可靠性,提出一种将图常量条件函数依赖转化为图修复规则的方法(GenGRR)。通过图模式在图中匹配同构子图并映射成节点-属性二维表,从表中相应属性域中抽取错误模式把图常量条件函数依赖转化成图属性值修复规则;删去图模式中常量条件函数依赖 RHS 对应的节点与相连边生成图属性补充规则。基于最大公共同构子图筛选并验证生成图修复规则的一致性。在多个真实数据集上进行测试,验证相比图常量条件函数直接修复图数据,通过转化生成的图修复规则具有更好的修复效果。

关键词:数据一致性;数据质量;图函数依赖;图修复规则;子图同构;最大公共同构子图
中图分类号:TP301 **文献标识码**:A **文章编号**:1673-629X(2024)04-0007-09
doi:10.20165/j.cnki.ISSN1673-629X.2024.0002

Graph Repairing Rule Discovery Based on Graph Constant Conditional Functional Dependencies

LI Jie[1,2,3],CAO Jian-jun[2,3],WANG Bao-wei[1],ZHUANG Yuan[1,2,3]
(1. School of Computer & Software,Nanjing University of Information Science and Technology,
Nanjing 210044,China;
2. The 63rd Research Institute,National University of Defense Technology,Nanjing 210007,China;
3. Laboratory for Big Data and Decision,National University of Defense Technology,Changsha 410073,China)

Abstract:Data consistency is an important part of data quality management. In order to improve graph data consistency,a lot of data dependency theories in relational database have been introduced into graph database,including graph functional dependencies,graph association rules and so on. Graph repairing rule is a newly proposed data dependency rule for graph with powerful repairing capability,but there is no effective mining algorithm yet. In order to automatically generate graph repairing rule and improve the reliability of graph data repairing,a method called GenGRR is proposed to transform graph constant conditional functional dependencies into graph repairing rules. By using the graph pattern,the isomorphic subgraph is matched and mapped into a node-attribute two-dimensional table,and the error pattern is extracted from the corresponding attribute field in the table to transform the constant condition function dependency into the graph attribute value repair rule. The graph attribute supplement rules are generated by deleting the nodes and contiguous edges of constant condition function dependent on RHS in graph mode. Based on the maximum common isomorphic subgraph,the consistency of the repair rules of the generated graph is screened and verified. It is tested on multiple real data sets to verify that the graph repair rule generated by transformation has better repair effect than that of the graph constant condition function.

Key words:data consistency;data quality;graph functional dependency;graph repairing rule;subgraph isomorphism;maximum common isomorphism subgraph

收稿日期:2023-08-11 **修回日期**:2023-12-13
基金项目:国家自然科学基金资助项目(61972207);中国博士后科学基金特别资助项目(2015M582832);国家重大科技专项(2015ZX01040201-003)
作者简介:李 杰(1998-),男,硕士研究生,研究方向为数据质量控制与数据治理;通信作者:曹建军(1975-),男,副研究员,博士,研究方向为数据质量控制与数据治理;王保卫(1982-),男,教授,博导,博士,研究方向为物联网数据安全、可信数据交易、信息隐藏、数字水印。

0 引 言

随着大数据时代来临,网络技术不断发展,图数据在诸多领域展现其重要性。数据规模增加往往意味着不可避免的数据质量问题,图数据同样充满了错误、缺失、冗余等数据质量问题,低质量的数据往往严重影响其实用价值[1]。数据一致性(Data Consistency)是数据质量领域的重要研究内容。数据一致性是指在数据集合中每个信息都不包含语义错误或相互矛盾的数据[2]。在图数据库中,同样存在着广泛的数据不一致问题,包括数值冲突、数据缺失等。数据修复在关系数据库和 XML 数据库中得到了广泛研究,许多基于可靠规则修复关系型数据库的语义和算法被提出,例如函数依赖(Functional Dependency)[3]、拒绝约束(Denial Constraints)[4]、修复规则(Fixing Rule)[5]等。

关系型数据中函数依赖有着丰富的研究及多种实用扩展[6]。随着图数据应用规模的不断扩大,许多学者将函数依赖理论引入图数据库[7-13]。图函数依赖(Graph Functional Dependencies, GFD)是函数依赖在图数据中基于图拓扑结构的扩展形式,表示图结构中节点之间蕴藏的数据依赖。相比其他图数据质量规则,如图健[14]与图关联规则[15],图依赖由于其强大的语义表达能力与可扩展性,被认为在数据质量领域具有极大的研究价值[16]。

文献[8-9]针对 RDF 图提出基于路径模式表示的图函数依赖,形式为 $\delta: V_n(X \rightarrow Y)$,以节点 V 及其 n 代子节点构建的子图决定拓扑结构,$X \rightarrow Y$ 决定函数依赖关系,并基于图-表-图的流程挖掘图函数依赖。首先以各节点为中心得到包含 n 层子节点的子图,以边标签为属性将子图转化成相应的表,用频繁模式挖掘算法从各个表中发现频繁模式,最后再用 CFDMiner 算法发现图条件函数依赖。以路径模式表示图结构因只能表达星型结构而具有天然局限性[16]。文献[10]提出以图模式表示图函数依赖的图结构。形为 $\varphi: Q[\mu](X \rightarrow Y)$,通过图模式 Q 指定拓扑结构避免了路径模式表示带来的不足。文献[11]在文献[10]的基础上,提出通过构建生成树挖掘图函数依赖,算法分为纵向拓展与横向拓展,纵向拓展自底向上生成所有规模小于 k 的图模式;横向拓展基于图模式匹配数据,依次生成图函数依赖。针对不同数据质量问题的图函数依赖的扩展被相继提出,包括解决数值不一致的数值图依赖[12]以及针对时序知识图谱的时序图依赖[13]等。文献[17]提出了图修复规则(Graph Repairing Rule, GRR),其形式为 $\varphi: (Q, X) \rightarrow (Q', Y)$,包括修复前后的图模式 Q,Q' 以及相应的字面量集合 X,Y。图修复规则被验证可以有效解决图数据中包括冲突、不完整、冗余在内的多种数据不一致问题,相比已有规则修复能力更为全面。文献[18]进一步提出了宽松图修复规则,放宽对包含错误数据子图的匹配限制,提高了修复的查全率。

数据质量的提升包括检测与修复两个方面,图函数依赖在图数据错误检测方面相比以往的规则具有更高的准确率[11]。但将其用于图数据的修复,查准率较低,较低的查准率可能反而导致数据质量的进一步下降。且图函数依赖仅能处理捕获子图的数据一致性问题,无法处理图结构变化相关的修复[18]。GRR 是最新提出的一种针对图数据的修复规则,具有强大的修复能力,但其目前尚无有效挖掘算法,需借由专家指定或人工推导,而图函数依赖目前已有成熟的挖掘算法。鉴于两者具有形式上相似及语义上相近的特征,该文的工作如下:

(1)提出了将图常量条件函数依赖转化为 GRR 的方法(GenGRR),一定程度上解决图修复规则无法自动挖掘的问题。

(2)基于转化提高了针对属性值修复时图常量条件函数依赖的查准率,提高了修复的可靠性。

(3)通过转化一定程度上解决了图常量条件函数依赖无法补充图数据缺失属性的问题。

(4)基于转化将生成的 GRR 用于图数据的修复,经过实验验证了修复的有效性。

1 相关概念

定义 1 条件函数依赖[6]。一个条件函数依赖记作:$\varphi: (X \rightarrow Y, t_p[X \| Y])$。$X \rightarrow Y$ 是标准函数依赖;t_p 为 $X \cup Y$ 上的某个具体的取值模式。对于属性 $A \in (X \cup Y)$,$t_p[A]$ 为变量或 Dom(A)中的某个特定常量。决定模式 $t_p[X]$ 和被决定模式 $t_p[Y]$ 一般称为条件函数依赖的 LHS 和 RHS,“ ‖ ”为 LHS 和 RHS 的分隔符。当 $[X \cup Y]$ 全部为常量值的时候,其为常量条件函数依赖。

定义 2 图(Graphs)。有向图数据被定义为 $G = (V, E, L, F)$,V 为有限节点集合,$E \subseteq V \times V$ 为边集合。每个节点 $v \in V$ 与边 $e \in E$ 都有一个表示其属性的标签 $L(v) \in L$ ($L(e) \in V$)。对每一个节点 v,存在函数 $f(L(v)) \in F$ 对其属性映射一个常量值。一个表示城市及其邮编地址的图数据如图 1 中 G_1 所示。

定义 3 图模式(Graph Patterns)[10]。图模式被定义为有向图 $Q = (V_Q, E_Q, L_Q, g)$。V_Q 为节点集合,$E_Q \subseteq V_Q \times V_Q$ 为边集合,对每个节点 $v \in V_Q$ 与边 $e \in E_Q$,$L_Q(v)$ 与 $L_Q(e)$ 表示其属性标签,g 是映射函数。典型图模式如图 1 中 $Q_1 \sim Q_3$ 所示。

定义 4 公共同构子图(Common Isomorphic Subgraph)。给定图 G_1 与 G_2,如果图 G_c 是 G_1 的子图,

且与 G_2 子图同构,称 G_c 为公共同构子图,如果从 G_1 中给 G_c 添加任何更多节点或边, G_c 不再与 G_2 子图同构,称 G_c 为最大公共同构子图。

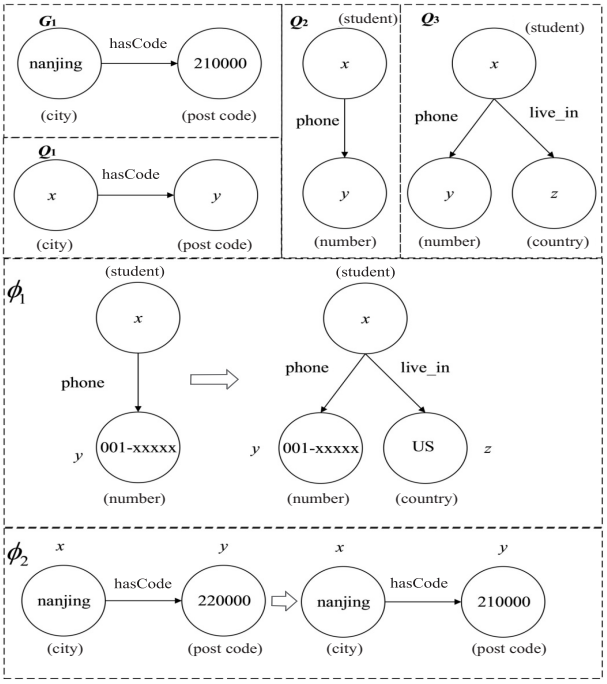


图 1　图,图模式及 GRR

定义 5 模式同构(Pattern Isomorphic)[16]。给定图 $G=(V, E, L, F)$ 和 $Q=(V_Q, E_Q, L_Q, g)$,若存在一个双射函数 $f: V \to V_Q$ 满足以下条件:

(1) $\forall u,v \in V, (u,v) \in E \Rightarrow (f(u),f(v)) \in E_Q$

(2) $\forall u,v \in E, L((u,v))=L_Q((f(u),f(v)))$

则称 G 与 Q 模式同构。模式同构问题可被视为图同构问题的子类问题,指在不考虑节点标签取值时的图同构,图 1 中 G_1 与 Q_1 模式同构。

定义 6 超图。如果图 g 是图 g' 的子图且与 g' 子图同构,那么称 g' 是 g 的一个超图。

定义 7 图模式匹配。给定图 G 和图模式 q,找出图 G 中所有与 q 模式同构的子图。

定义 8 图常量条件函数依赖。一个图常量条件函数依赖记作 $\varphi: Q[\mu](X \to Y, t_p[X \| Y])$。$Q[u]$ 是图模式, $(X \to Y, t_p[X \| Y])$ 是标准常量条件函数依赖。与关系型数据中的常量条件函数依赖不同, X 与 Y 为字面量集合,字面量描述指定节点及其对应的取值。该文考虑图常量条件函数依赖的一般形式 $\varphi: Q[\mu](X \to l, t_p[X \| l])$,其中 l 为单个字面量, $Q[\mu](X \to Y)$ 可等价于一系列 $Q[\mu](X \to l)$ 集,其中 $l \in Y$[11]。

$\varphi: Q_1[u](x \to y, (x.\text{val} = \text{nanjing} \| y.\text{val} = 210000))$ 为一个典型的图常量条件函数依赖, 图模式如图 1 中 Q_1 所示,LHS:x. val = nanjing, RHS:y. val = 210000,其含义为当城市节点 x 为‘nanjing’时,唯一决定其邮编节点 y 为‘210000’。

定义 9 非平凡图函数依赖(Nontrivial GFDs)[11]。一个图函数依赖 $\varphi: Q[u](X \to l)$,如果 X 非‘false’且 l 不能由 X 等价转换,则称其为非平凡图函数依赖。与关系型数据类似,图函数依赖挖掘关心的是非平凡图函数依赖,该文默认所有图依赖皆为非平凡图函数依赖。

定义 10 图修复规则[18]。$\varphi: (Q, X) \to (Q', Y)$,其中 Q 与 Q' 表示修复前与修复后的图模式, X 与 Y 分别表示描述图模式 Q 与 Q' 的字面量集合。文献[18]将图数据的修复划分为五种类型:属性补充、关系补充、关系分辨、属性值修复、实体分辨,该文主要考虑属性值修复与属性补充。

两个 GRR $\varphi_1: (Q_2, X_1) \to (Q_3, Y_1)$ 与 $\varphi_2: (Q_1, X_2) \to (Q_1, Y_2)$ 如图 1 所示,其中 Q_1, Q_2, Q_3 分别如图 1 中 Q_1, Q_2, Q_3 所示。φ_1 中 $X_1 = \{y.\ \text{val} = \text{“001-xxxxx”}\}$, $Y_1 = \{y.\ \text{val} = \text{“001-xxxxx”}, z.\ \text{val} = \text{“US”}\}$。$\varphi_1$ 表示如果某人电话号码为 001 开头,那么可补充出此人居住地址为美国的信息。φ_2 中 $X_2 = \{x.\ \text{val} = \text{“nanjing”}, y.\ \text{val} = \text{“220000”}\}$, $Y_2 = \{x.\ \text{val} = \text{“nanjing”}, y.\ \text{val} = \text{“210000”}\}$。$\varphi_2$ 表示若“nanjing”的邮编为“220000”,则需要将其修改成“210000”。显然, φ_1 属于属性补充 GRR, φ_2 属于属性值修复 GRR。

定义 11 GRR 一致性[17]。给定一组 GRRΣ,一致性问题在于判断是否存在一个非空图 G 能满足 Σ 中所有 GRR。即给定两个 GRR,如果二者具有一致性,那么它们修复相同的边和节点,结果是相同的。

2　转化框架

基于定义发现,对比图常量条件函数依赖与 GRR $\varphi: (Q, X) \to (Q', Y)$,两者都由图模式与描述图模式的字面量集合构成。图常量条件函数依赖 LHS 可映射为 GRR 中 X 与 Y 的子集,RHS 可映射为 Y 的子集,区别在于 X 中需要包含错误数值或 Q 中包含错误结构。通过在 X 中添加错误数值可生成修复属性值的 GRR。常量条件函数依赖可视为置信度 100% 的关联规则,具有一定的缺失数据补充能力,即通过删去 RHS 属性值对应的节点与相连边作为待修复的图模式可转化为补全缺失属性的 GRR。图常量条件函数依赖可转变为 2 种不同类型的 GRR:(1)属性值修复;(2)属性补充。图常量条件函数依赖转换生成 GRR 的流程如图 2 所示。

首先基于现有算法在图数据 G 上挖掘图常量条件函数依赖非平凡集[11]。按照一定标准筛选出符合要求的图常量条件函数依赖集 Σ_{GFD}(例如非冗余,一致集或支持度较高的图依赖)。随后从图常量条件函数依赖集 Σ_{GFD} 分割出图模式集 Σ_{GP},通过图模式集在 G

中进行图匹配得到二维映射表集 Σ_T(为了之后抽取错误数值)。基于算法 GenGRR 生成 GRR 初始集 Σ_{GRR} 并检查其一致性,如果初始集具有一致性,则输出结果,如果存在不一致规则,则检查处理 GRR 初始集 Σ_{GRR} 中的不一致规则,修改或删去部分 GRR 直到满足一致性。

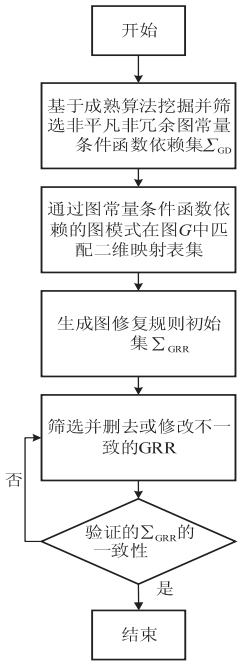


图 2 流程图

3 规则转化

3.1 映射二维表生成

为了生成 GRR,需得到相应的二维映射表。首先根据图模式匹配图中所有模式同构子图,之后将节点根据位置映射成二维表。其过程如图 3 所示。

图模式匹配过程如算法 1 所示。

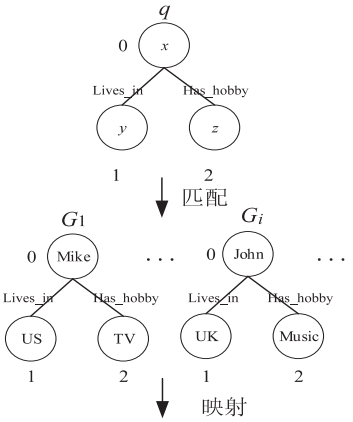


图 \ 相对位置	0	1	2
G_1	Mike	US	TV
…	…	…	…
G_i	John	UK	Music
…	…	…	…

图 3 图模式匹配过程

算法 1:图模式匹配 Patternmatching

输入:图常量条件函数依赖初始集 Σ_{GFD},图数据 G。

输出:二维映射表集 Σ_T。

(1) 初始化:二维映射表集 $\Sigma_T = \varnothing$

(2) Σ_{GFD} 拆分出无重复的图模式集 Σ_{GP}

(3) For 图模式 q In Σ_{GP}

(4) Σ_{Sub} =GraphMatching(G,q) // 匹配同构子图

(5) T =RationaltableIndex(Σ_{Sub}) // 映射成二维表

(6) $\Sigma_T = \Sigma_T \cup T$

(7) Return Σ_T

算法第 2 行将依赖集拆分出不重复的图模式集 Σ_{GP}。第 4 行中方法 GraphMatching 找到图模式 q 在图 G 中所有模式同构的子图集 Σ_{Sub},方法 RationaltableIndex 将所有同构子图的节点按照固定顺序映射成节点-属性二维表。

现有图匹配技术多采用基于索引的思想在图中针对有效特征建立倒排索引,以减少搜索空间[19]。其过程包括索引建立以及模式匹配。精确图匹配是 NP 难问题,随着问题规模的增长,匹配时间将呈指数级增长[20],但由于挖掘的图函数依赖图模式规模 k(节点数量)上限往往设置在 6 以内[11]。精确图模式匹配算法能在可接受的时间范围内生成映射二维表。

将图模式视为边的集合,首先根据边标签建立倒排索引并引入 DFS 编码,减少模式匹配时间。DFS 编码是解决子图同构的有效方法,可将一个图转变为边的序列。如果两个图的最小 DFS 编码相同,那么它们是图同构的。在已知图模式的情况下,无需计算最小 DFS 编码,通过边的添加顺序保证编码唯一性,以判断子图是否同构。图匹配方法 GraphMatching 如算法 2 所示。

算法 2:图模式匹配算法 GraphMatching

输入:图数据 G,图模式集 q

输出:模式同构的子图集 Σ_{Sub}

(1) 初始化:图模式 q 对应边集 Σ_l,图模式 q 对应 DFS 编码序列 C,边标签索引库 $\Sigma_{index} = \varnothing$,匹配子图集 $\Sigma_{Sub} = \varnothing$

(2) For l In Σ_l

(3) $\Sigma_{index} = \Sigma_{index} \cup$ GetIndex(G,l) //构建边索引集合

(4) End For

(5) $C' = C$

(6) $c = C'$. pop( ) //得到图模式 q 中第一条边对应的 DFS 码

(7) Σ_{Sub} =GetEdge(c , Σ_{index}) //根据 DFS 码得到相应边集合

(8) s =0 //控制构建超图的次数

(9) While($s < |\Sigma_l|$)

(10) $c = C'$. pop( ) //依次得到图模式 q 的 DFS 码

(11) Σ_l^c = GetEdge(c , Σ_{index}) //根据 DFS 码得到相应边集合

(12)　For l in Σ_l^c

(13)　GenerateSupergraphs(Σ_{Sub} , l)　//根据索引生成超图

(14)　$s = s + 1$

(15) For 子图 g In Σ_{Sub}

(16)　If GetDFScode(g)! = C //通过 DFS 码判断是否模式同构

(17)　$\Sigma_{Sub} = \Sigma_{Sub} / g$

(18) Return Σ_{Sub}

算法 2-4 行建立边索引库。算法 5-7 行首先匹配规模为 2 的子图集 Σ_g ,即含有两个节点与一条边的子图。第 8-9 行通过 s 记录循环次数,不断添加新的边构建超图。第 10-11 行通过图模式 q 边的 DFS 编码依次找到相应的边集 Σ_l^c 。第 12-14 行通过 Σ_l^c 中的边构造子图集 Σ_g 的超图。第 15-17 行通过 DFS 编码判断,保留 Σ_g 中与图模式同构的子图。

3.2　转化生成算法

图常量条件函数依赖转化成图修复规则算法 GenGRR 如算法 3 所示。

算法 3:转化算法 GenGRR

输入:图数据 G ,图模式集 Σ_{GP} ,二维映射表集 Σ_T

输出:图修复规则集 Σ_{GRR}

(1) 初始化:GRR 集 $\Sigma_{GRR} = \varnothing$,属性值修复 GRR $\Sigma_{C-GRR} = \varnothing$,属性补充 GRR 集 $\Sigma_{S-GRR} = \varnothing$

(2) Σ_{C-GRR} =GenC_GRRs(Σ_{GP} , Σ_T)

(3) Σ_{S-GRR} =GenS_GRRs(Σ_{GP})

(4) $\Sigma_{GRR} = \Sigma_{C-GRR} \cup \Sigma_{S-GRR}$

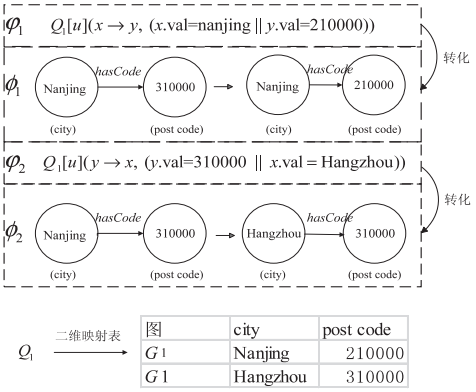
(5) Σ_{GRR} =InconsistenceDetect(Σ_{GRR})

(6) ReturnΣ_{GRR}

算法 2-3 行分别通过方法 GenC_GRRs 与 GenS_GRRs 生成属性值修复 GRR 与属性补充 GRR,第 5 行对生成的 GRR 进行一致性检测,最终返回生成的 GRR 集合。

3.3　属性值修复 GRR 生成

为方便叙述,设属性值修复 GRR: (Q, X) → (Q, Y) 字面量集合 X 与 Y 的交集 $X \cap Y$ 为证据模式 A 。$X \setminus A$ 为错误模式 B , $Y \setminus A$ 为事实模式 C ,修复前后图模式相同, X . n 表示 X 描述的节点集, X . v 表示所描述节点的取值集。以第一节中属性值修复 GRR: φ_2 为例, X_2 = { x . val = "nanjing", y . val = "220000" }, Y_2 = { x . val = "nanjing", y . val = "210000" } ,则节点集 X_2. n = { x , y } ,取值集 X_2. v = {"nanjing","220000"} ,证据模式 A = { x . val = "nanjing" } ,错误模式 B = { y . val = "220000" } ,事实模式 C = { y . val = "210000" } 。显然,错误模式与事实模式描述相同节点,但节点值不同,即 B . n = C . n, B . v ≠ C . v。两个图常量条件函数依赖与生成的属性值修复 GRR 及图模式 Q_1 对应二维映射表如图 4 所示,图模式 Q_1 如图 1 所示。



Q_1 二维映射表

图	city	post code
$G1$	Nanjing	210000
$G1$	Hangzhou	310000

图 4　规则转化

属性值修复 GRRϕ_1 = (Q_1 , X) → (Q_1 , Y) ,修复前后图模式与 φ_1 相同,LHS: x . val = Nanjing 映射为 ϕ_1 的证据模式,RHS: y . val = 210000 映射为 ϕ_1 的事实模式, Q_1 二维映射表 RHS 描述节点(post code)除事实模式取值以外的值映射为 ϕ_1 的错误模式取值。

二维映射表中抽取错误模式可生成属性值修复 GRR,同理 φ_2 可转化为 ϕ_2 。显然, ϕ_1 与 ϕ_2 违反 GRR 一致性,修复相同子图时将得到冲突结果。直接添加错误模式生成的 GRR 集冲突严重,后续修复困难。为保证一致性,需对添加的错误模式进行必要约束。

给定属性值修复 GRRφ 与 ϕ ,两者图模式相同。φ 证据模式为 A ,错误模式为 B ,事实模式为 C 。ϕ 证据模式为 X ,错误模式为 Y ,事实模式为 Z 。φ 与 ϕ 产生冲突的情况有如下四种:(1)若 Z . n = C . n,当 Y. v ∩ B. v ≠ ∅ , Z. v ≠ C. v 时冲突。(2)若 Z . n ≠ C . n, Z. n ∈ A. n , C. n ∉ X. n ,当 (Z. n ∩ A. n). v ∈ Y. v 时冲突。(3)若 Z . n ≠ C . n, Z. n ∉ A. n , C. n ∈ X. n ,当 (C. n ∩ X. n). v ∈ B. v 时冲突。(4)若 Z . n ≠ C . n, Z. n ∈ A. n , C. n ∈ X. n ,当 (C. n ∩ X. n). v ∈ B. v 且 (Z. n ∩ A. n). v ∈ Y. v 时冲突。

根据给出的四种冲突情况,转化具有相同图模式的图常量条件函数依赖时,需要将新转化的 GRR 与已生成的 GRR 进行对比判断,删去可能导致冲突的错误模式取值以保证一致性,转化方法如算法 4 所示。

算法 4:GenC_GRRs

输入:图常量条件函数依赖集 Σ_{GFD} ,二维映射表集 Σ_T

输出:属性值修复 GRR 集 Σ_{C-GRR}

(1) 初始化属性值修复集 $\Sigma_{C-GRR} = \varnothing$

(2) Σ_{GFD} 拆分出无重复的图模式集 Σ_{GP}

(3) For 图模式 q In Σ_{GP}

(4) $\Sigma_{C-GRR}^q = \varnothing$ //初始化图模式为 q 的属性值修复 GRR 集

(5)　T =getTable(Σ_T , q) //找到图模式 q 对应的映射二维表

(6)　Σ_{GFD}^q =getGFD(Σ_{GFD} , q) //找到图模式为 q 的图常量条件函数依赖集 Σ_{GFD}^q

(7) For φ^f:$Q_f[\mu](X_f \rightarrow Y_f, t_p[X_f \| Y_f])$ In Σ^q_{GFD}

(8) 初始化属性值修复 GRR: φ^R,其修复前后图模式为 Q_f,证据模式为 X,错误模式为 Y,事实模式为 Z

(9) Y = getDom(T, l.n)\ l.v //从映射表 T 中得到 l 描述节点对应的属性域除去 l 取值并映射为 $\emptyset$ 的错误模式。

(10) $X = X_f$, $Z = Y_f$ // φ^f 中 X_f 赋予 X, Y_f 赋予 Z

(11) For 属性值修复 GRR: $\emptyset^R$ In Σ^q_{GRR}

(12) $\emptyset^R$ 证据模式为 A,错误模式为 B,事实模式为 C

(13) If X.n ∩ Y.n = ∅ or X.n ∩ Y.n = w, (X.n ∩ w).v = (Y.n ∩ w).v

(14) If Z.n = C.n and Z.v ≠ C.v //第一种冲突情况

(15) Y.v = Y.v\B.v //删去可能导致冲突的错误模式取值

(16) End If

(17) If Z.n≠ C.n, Z.n ∈ A.n, C.n ∉ X.n //第二种冲突情况

(18) Y.v = Y.v\(Z.n ∩ A.n).v //删去可能导致冲突的错误模式取值

(19) End If

(20) If Z.n≠ C.n, Z.n ∉ A.n, C.n ∈ X.n //第三种冲突情况

(21) B.v = B.v\(C.n ∩ X.n).v //删去可能导致冲突的错误模式取值

(22) End If

(23) If Z.n≠ C.n, Z.n ∈ A.n, C.n ∈ X.n //第四种冲突情况

(24) B.v = B.v\(C.n ∩ X.n).v Y.v = Y.v\(Z.n ∩ A.n).v

(25) End If

(26) End If

(27) End For

(28) $\Sigma^q_{C-GRR} = \Sigma^q_{C-GRR} \cup \varphi^R$

(29) End for

(30) $\Sigma_{C-GRR} = \Sigma_{C-GRR} \cup \Sigma^q_{C-GRR}$

(31) End for

(32) Return Σ_{C-GRR}

第 3 行对图模式遍历,优先处理转化中具有相同图模式 GRR 的冲突情况,不同图模式 GRR 间的冲突检测方法在 3.5 节讨论。4-6 行找到转化的必要元素,包括图模式为 q 的图常量条件函数依赖以及 q 的二维映射表。第 7 行对图模式为 q 的图常量条件函数依赖 φ^f 遍历依次转化,8-10 行初始化生成的 GRR,将 φ^fLHS 映射为 GRR 证据模式,RHS 映射为事实模式,映射表中 RHS 描述结点取值域除去事实模式取值以外的值作为 GRR 错误模式取值。第 11 行对已生成的图模式为 q 的 GRR 循环遍历,依次与新生成的 GRR 对比,第 13 行对给定的两个 GRR 进行判断,仅当两者的证据模式描述不同节点,或描述的共同节点取值相同时,才会对同一实体产生共同作用,即可能出现冲突情况。14-24 行根据四种可能产生的冲突情况进行判断,删去可能导致冲突的错误模式取值。31-32 行返回生成的 GRR 集。

3.4 属性补充 GRR 生成

图常量条件函数依赖仅关注图数据中节点属性值的数据一致性,对缺失属性的修复无能为力[18]。删去图模式中常量条件函数依赖 RHS 属性值对应的节点与所有相连边并作为修复前的图模式 Q,原图模式为修复后的图模式 Q',可生成属性补充 GRR 集 Σ_{S-GRR},生成过程如算法 5 所示。

算法 5:GenS_GRRs

输入:图常量条件函数依赖一致集 Σ_{GFD}

输出:属性补充 GRR 集 Σ_{S-GRR}

(1) 初始化图模式集 $\Sigma_{S-GRR} = \emptyset$

(2) For 图常量条件函数依赖 φ^f:$Q_f[\mu](X_f \rightarrow Y_f, t_p[X_f \| Y_f])$ In Σ_{GFD}

(3) 初始化属性补充 GRR: $\emptyset$ 证据模式 X,事实取值 Z,修复后图模式 Q',修复前图模式 Q

(4) $X = X_f$, $Z = Y_f$, $Q' = Q_f$

(5) 删去 Q_f 中 Y_f 对应节点与所有相连边得到 Q_d

(6) $Q = Q_d$

(7) $\Sigma_{S-GRR} = \Sigma_{S-GRR} \cup \emptyset$

(8) End For

(9) Return Σ_{S-GRR}

算法第 2 行将所有图常量条件函数依赖 φ^f 依次转化为属性补充 GRR$\emptyset$,第 4 行中分别将 φ^f 的 LHS 与 RHS 映射为 $\emptyset$ 的证据模式与事实模式,φ^f 的图模式 Q_f 映射为 $\emptyset$ 修复后的图模式 Q',属性补充 GRR 补充图数据中缺失的属性,没有错误模式。第 5 行将图模式 Q_f 中 RHS 对应的节点以及与节点相连边删去作为 $\emptyset$ 修复前的图模式 Q,Q 可为不连通的图模式。第 7-8 行返回生成属性补充集 Σ_{S-GRR}。

3.5 一致性验证

3.3 节中通过避免四种冲突情况可保证同一图模式下生成 GRR 集的一致性,但是无法保证不同图模式下 GRR 的一致性,需要进一步通过验证及筛选保证整体 GRR 集的一致性。

定理 1:给定两个 GRR, $\emptyset_1 = (Q_1, X_1) \rightarrow (Q_1', Y_1)$ 以及 $\emptyset_2 = (Q_2, X_2) \rightarrow (Q_2', Y_2)$,图 Q_c 是 Q_1 与 Q_2 的最大公共同构子图,如果 Q_c 非空,在经过 $\emptyset_1$ 与 $\emptyset_2$ 的修复之后,Q_c 被变为 Q_A 与 Q_B。设 $X_A \subseteq X_1$, $Y_A \subseteq Y_1$, $X_B \subseteq X_2$, $Y_B \subseteq Y_2$ 表示 Q_c 在 $\emptyset_1$ 与 $\emptyset_2$ 中各自对应的字面量集。如果 $\emptyset_1$ 与 $\emptyset_2$ 是一致的,那么应满足:(1) Q_A 与 Q_B 是同构的;(2) $X_A = X_B$;(3) $Y_A = Y_B$。

证明:如果 GRR$\emptyset_1$ 与 $\emptyset_2$ 是一致的,那么对同一节点或边修复结果是相同的。最大公共同构子图 Q_c 表

示 ϕ_1 与 ϕ_2 共同修复的子图部分。(1)确保共同修复部分修复后结构相同;(2)确保修复前字面量集合相同,即满足修复条件;(3)确保修复后的字面量集合相同,即修复的结果取值相同。

推论 1:当且仅当 GRR 集 Σ_{GRR} 中的每一对 GRR 是一致的,则该 GRR 集是一致的。

基于定理 1 与推论 1 筛选 GRR 一致集方法如算法 6。

算法 6:InconsistenceDetect
输入:GRR 集 Σ_{GRR}
输出:GRR 冲突集 Σ_{conf}
(1) For Σ_{GRR} 中每一对图模式不同的 GRRϕ_1 与 ϕ_2
(2)　$\phi_1 = (Q_1, X_1) \rightarrow (Q_1', Y_1)$;
(3)　$\phi_2 = (Q_2, X_2) \rightarrow (Q_2', Y_2)$;
(4)　Q_c = findcommon(Q_1,Q_2);
(5)　If Q_c 非空
(6)　 If $X_A = X_B \&\& Y_A \neq Y_B$
(7)　　Put $\{\phi_i, \phi_j\}$ In Σ_{conf}
(8)　 End If
(9)　End If
(10) End For

为了验证 GRR 集 Σ_{GRR} 的一致性,第 4 行对每一对图模式不同的 GRR 循环遍历,方法 findcommon 参考文献[21]中的方法找到 ϕ_i 与 ϕ_j 的最大公共同构子图 Q_c 并利用其验证一致性,算法 5-7 行找出非一致 GRR 对放入冲突集 Σ_{conf}。对找出的 GRR 冲突对随机删除其中之一或修改以满足一致性。

4　实验与分析

4.1　实验环境

实验基于 Ubuntu16.04 的服务器实现,CPU 型号为 Intel Xeon E5-2630,主存为 192 GB。算法基于 Python3.8 实现。

4.2　数据集及预处理

实验选取 2 个图数据集,真实数据集 FB15K 以及 WN18RR,二者均为知识图谱领域常用图数据集。FB15K 包含 14 951 个实体,1 345 种关系,50 000 个三元组,WN18RR 包含 40 943 个实体,11 种关系,86 835 个三元组。实验采用类似研究的配置[11]:以原始数据为干净数据,对图数据中 α% 个节点添加两种类型噪声,即属性值错误与属性缺失噪声,属性缺失包括节点缺失与边缺失。

4.3　评价指标

采用查准率(Precision)和查全率(Recall)作为评价指标,各指标计算公式分别如式 1 和式 2 所示。

$$\text{Recall} = \frac{|V^{GD} \cap V^{E}|}{|V^{E}|} \tag{1}$$

$$\text{Precision} = \frac{|V^{GD} \cap V^{E}|}{|V^{GD}|} \tag{2}$$

其中,V^E 代表引入的噪声,即代表实际违反图依赖的节点;V^{GD} 代表被图依赖所捕获并正确修复的节点。

4.4　实验结果与分析

设计 3 个实验评估规则转化算法 GenGRR:(1)验证输入的图常量条件函数依赖与生成图修复规则数量对比;(2)验证图常量条件函数依赖与转化的 GRR 修复效果对比;(3)验证噪声率以及转化数量对修复效果影响。

1)生成效率。

表 1 给出了在两个不同数据集上,不同的输入图常量条件函数依赖数量与生成的 GRR 的数量对比。

由表 1 可知,一条图常量条件函数依赖可生成一条属性值修复 GRR 与一条属性补充 GRR,因此生成的 GRR 数量以 1∶2 的比例增加。随着输入数量的增加,GRR 之间冲突加剧,当 FB15K 数据集上输入 500 条图常量条件函数依赖时,出现了由于违反一致性被删去了 3 条 GRR 的情况。

表 1　算法 GenGRR 生成的 GRR 结果

输入	FB15K	WN18RR
200	400	400
300	600	600
400	800	800
500	997	1 000

2)修复效果。

根据谷歌的调查,谷歌知识库包含约 20% 错误[18],因此在数据集中添加 20% × | V | 个噪声模拟真实数据,| V | 表示节点的数量,其中属性值错误和属性缺失噪声比例为 1∶1。输入图常量条件函数依赖数量为 500。由于随机添加噪声具有随机性,取 10 次实验结果求平均。实验对比方法包括目前图数据领域常用的数据依赖规则:图函数依赖(GFD)[11],图关联规则(Graph Association Rule,GAR)[15],GRR(基于 GFD 转化)[17],以及属性值图修复规则(C-GRR),属性补充规则(S-GRR)。

表 2　修复结果

approaches	FB15K		WN18RR	
	Precision	Recall	Precision	Recall
GFD	0.411	0.415	0.263	0.348
GAR	0.905	0.230	0.640	0.134
GRR	0.713	0.620	0.585	0.533
C-GRR	0.610	0.161	0.683	0.105
S_GRR	0.803	0.459	0.531	0.428

表 2 描述了在两个不同数据集下的修复结果。从

表中可知:相比转化前的 GFD,转化而来的 GRR 在查准率以及查全率上都更具优势,具体分析如下:

(1)由于 GFD 只能修复捕获子图中的错误节点值,而无法修复缺失属性,转化的 GRR 查全率更高。GRR 对于错误数据的匹配相比更为严格,因而查准率更高。

(2)比较属性值修复 C-GRR 以及 GFD 的修复能力,取得了关系型数据中 FR 的类似修复效果[22]。通过添加错误模式,提高对错误数据匹配要求,修复结果具有更高的查准率,但查全率较低。即 GFD 在错误检测任务中更为强大,而 GRR 在数据修复中更为‘可靠’。

(3)比较属性补充规则 S-GRR 与 GAR,由于 GAR 仅能补充部分缺失边,且节点中添加的噪声同样显著影响了 GAR 的缺失边补充能力,因此 GAR 的查全率较低。S-GRR 补充属性时会同时补充边与节点,相对查全率较高。

3)参数影响。

图数据修复过程中,噪声比例会对修复结果产生重要影响。设定输入的图常量条件函数依赖个数为 500,对图数据中添加 10% ~90% 的噪声,以此为实验验证效果,结果如图 5 和图 6 所示。

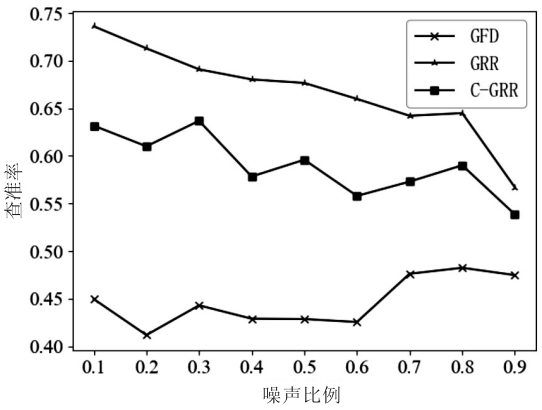


图 5 噪声比例对查准率的影响

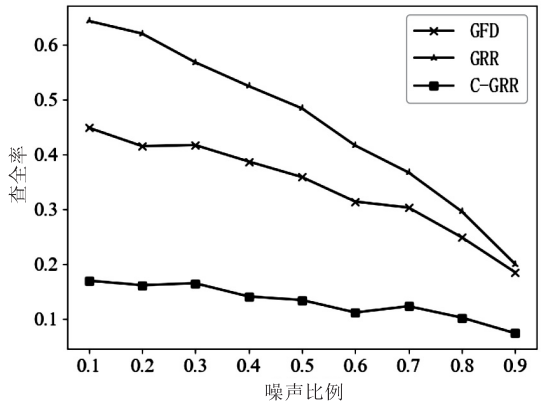


图 6 噪声比例对查全率的影响

当图数据中噪声比例上升时 GRR 查准率逐渐下降,图常量条件函数依赖查准率变化较小,最后略有上升。GRR 和图常量条件函数依赖查全率随着噪声比例上升都有明显的下降,但 GRR 下降幅度更大,这是由于 GRR 对错误数据的匹配更为苛刻,对噪声比例的上升更为敏感。文中 GRR 错误模式由映射二维表抽取而来,X 中抽取的错误模式数量决定了 GRR 的错误捕获能力。当输入的图常量条件函数依赖数量增加,生成的 GRR 冲突概率增加,抽取过程中以及验证一致性过程中为了保证 GRR 集的一致性,需删去导致冲突的部分错误模式,因此输入的图常量条件函数依赖数量对于转化生成 GRR 的修复能力同样具有重要影响。理论上错误模式初始数量为 | Dom(RHS. n) | -1,即 RHS 描述节点对应二维表属性的值域除去事实模式取值之后的数量。定义错误模式平均留存率(w_rate)来表示一个 GRR 的数据修复能力,如式 3 所示。

$$\text{w\_rate} = \frac{\sum_{i=1}^{|\text{C-GRR}|} \frac{|w_i|}{|\text{Dom(RHS. n)}| - 1}}{|\text{C - GRR}|} \tag{3}$$

式中,$|w_i|$ 表示第 i 个属性值修复 GRR 的错误模式数量,属性补充 GRR 只有图模式上的缺失,没有错误模式,不参与计算。设定噪声比例为 20%,输入的 GFD 数量与对应转化的 GRR 集的修复效果如图 7 所示。

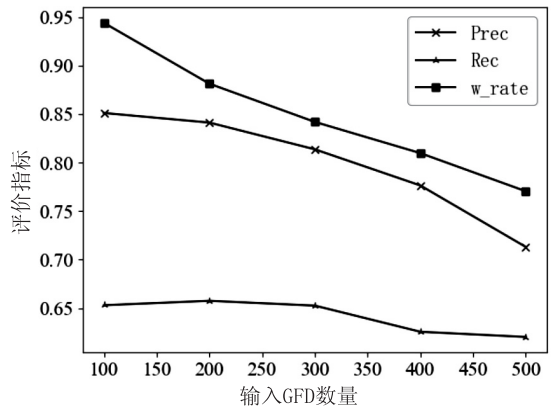


图 7 输入 GFD 对修复效果的影响

当输入的图常量条件函数依赖数量增加时,修复的查全率、查准率以及错误模式平均留存率都不断下降。显然,由于转化数量的增多,冲突情况加剧,被删去的错误模式比例不断增加,GRR 的修复能力也不断下降。

5 结束语

GRR 是最新提出的一种针对图数据的修复规则,可以有效解决多种图数据中的数据不一致问题。但 GRR 目前尚无有效的挖掘算法,需要由专家指定或由现有依赖人工推导,而图常量条件函数依赖已经有了多种成熟的挖掘算法。鉴于二者在结构与语义上都有着高度的相似性,该文提出了一种将图常量条件函数

依赖转化成图修复规则的有效方法,并在多个数据集上验证其修复效果的优越性。实验结果表明,相比通过图常量条件函数依赖直接修复图数据,转化生成的 GRR 可处理两种不同的图数据质量问题,因而在查全率与查准率都有明显的优势。仅针对图数据中节点值的数据不一致问题时,GRR 由于匹配条件更为苛刻,查准率更高,因而在修复方面则更为可靠。该文通过转化生成了两种修复类型的 GRR,但 GRR 理论可以修复图数据中五种不同类型的错误,未来工作主要集中在探究如何通过自动转化生成更多类型的 GRR 并修复图数据。

参考文献:

[1] LOSHIN D. 数据质量改进实践指南[M]. 曹建军,江　春,许　志,译. 北京:国防工业出版社,2016:10-12.

[2] 杜岳峰,李晓光,宋宝燕. 异构模式中关联数据的一致性规则发现方法[J]. 计算机研究与发展,2020,57(9):1939-1948.

[3] PAPENBROCK T, NAUMANN F. A hybrid approach to functional dependency discovery[C]//Proceedings of the 2016 international conference on management of data. San Francisco:ACM,2016:821-833.

[4] PENA E H M,DE ALMEIDA E C,NAUMANN F. Fast detection of denial constraint violations[J]. Proceedings of the VLDB Endowment,2021,15(4):859-871.

[5] WANG J,TANG N. Dependable data repairing with fixing rules[J]. Journal of Data and Information Quality,2017,8(3-4):1-34.

[6] LI M,WANG H,LI J. Mining conditional functional dependency rules on big data[J]. Big Data Mining and Analytics,2019,3(1):68-84.

[7] FAN Wenfei. Dependencies for graphs:challenges and opportunities[J]. Journal of Data and Information Quality,2019,11(2):1-12.

[8] YU Yang, HEFLIN J. Extending functional dependency to detect abnormal data in RDF graphs[C]//The semantic web - ISWC 2011:10th international semantic web conference. Bonn:Springer,2011:794-809.

[9] HE Binbin, ZOU Lei, ZHAO Dongyan. Using conditional functional dependency to discover abnormal data in RDF graphs[M]//Proceedings of semantic web information management on semantic web information management. Snowbird:ACM,2014:1-7.

[10] FAN W,LU P. Dependencies for graphs[J]. ACM Transactions on Database Systems,2019,44(2):1-40.

[11] FAN Wenfei,HU Chunming,LIU Xueli,et al. Discovering graph functional dependencies[J]. ACM Transactions on Database Systems,2020,45(3):1-42.

[12] FAN W,LIU X,LU P,et al. Catching numeric inconsistencies in graphs[J]. ACM Transactions on Database Systems,2020,45(2):1-47.

[13] NORONHA L, CHIANG F. Discovery of temporal graph functional dependencies[C]//Proceedings of the 30th ACM international conference on information & knowledge management. [s. l.]:ACM,2021:3348-3352.

[14] ANGLES R,BONIFATI A,DUMBRAVA S,et al. Pg-keys:keys for property graphs[C]//Proceedings of the 2021 international conference on management of data. [s. l.]:ACM,2021:2423-2436.

[15] WANG X,XU Y,ZHAN H. Extending association rules with graph patterns[J]. Expert Systems with Applications,2020,141:112897.

[16] 余　旭,曹建军,翁年凤,等. 图依赖研究与应用综述[J]. 计算机应用研究,2023,40(5):1312-1317.

[17] CHENG Yurong,CHEN Lei,YUAN Ye,et al. Rule-based graph repairing:semantic and efficient repairing methods[C]//2018 IEEE 34th international conference on data engineering (ICDE). Paris:IEEE,2018:773-784.

[18] CHENG Yurong,CHEN Lei,YUAN Ye,et al. Strict and flexible rule-based graph repairing[J]. IEEE Transactions on Knowledge and Data Engineering,2022,34(7):3521-3535.

[19] 项英倬,谭菊仙,韩杰思,等. 图匹配技术研究[J]. 计算机科学,2018,45(6):27-31.

[20] 于　静,刘燕兵,张　宇,等. 大规模图数据匹配技术综述[J]. 计算机研究与发展,2015,52(2):391-409.

[21] MCCREESH C,PROSSER P,TRIMBLE J. A partitioning algorithm for maximum common subgraph problems[C]//26th international joint conference on artificial intelligence. Melbourne:IJCAI Organization,2017:712-719.

[22] ZHOU Jinling,DIAO Xinchun,CAO Jianjun,et al. A method for generating fixing rules from constant conditional functional dependencies[C]//2016 IEEE international conference on knowledge engineering and applications (ICKEA). Singapore:IEEE,2016:6-11.