

规则库的合并以及冗余、环路和冲突的检测

贾秀芳 宋良图 王儒敬 李伟

(中国科学技术大学自动化系 安徽 合肥 230026)
(中国科学院合肥智能机械研究所 安徽 合肥 230031)

摘要 针对知识库的建立需要耗费大量的时间和人力,同时相同或相似领域的知识库数量越来越多,提出利用现有规则知识库进行合并生成一个新的规则知识库,并对生成的新规则知识库进行知识冗余、环路和冲突的检测算法。首先,规则库利用有向超图来表示;其次,将有向超图利用其邻接矩阵来表示,那么规则库的合并可以转换成有向超图所对应的邻接矩阵的合并,并依据邻接矩阵求可达矩阵以及利用总可达矩阵来检测规则库中规则的冗余、环路和冲突。最后,算法的有效性通过实例加以验证。

关键词 规则库 规则库合并 有向超图 邻接矩阵 可达矩阵

中图分类号 TP3 文献标识码 A DOI:10.3969/j.issn.1000-386x.2016.01.061

RULE BASES MERGING AND REDUNDANCY, CIRCULARITY AND CONFLICT DETECTION

Jia Xiufang Song Liangtu Wang Rujing Li Wei

(Department of Automation, University of Science and Technology of China, Hefei 230026, Anhui, China)
(Institute of Intelligent Machines, Chinese Academy of Sciences, Hefei 230031, Anhui, China)

Abstract To build the knowledge base needs to spend a lot of time and manpower, meanwhile there are more and more knowledge bases built in the same or similar fields. In light of this, we proposed an algorithm, which utilises existing rule knowledge bases and merges them to generate a new rule knowledge base, and detects the knowledge redundancy, circularity and conflict in the generated new rule knowledge base. First, we represented the rule base by the directed hypergraph; secondly, we represented the directed hypergraphs by their adjacency matrices, thus the merging of rule bases can be converted into the merging of the corresponding adjacency matrices of the directed hypergraphs, we then obtained the reachability matrix according to adjacency matrices, and used the total reachability matrix to detect the redundancy, circularity and conflict of rules in rule base. Finally, the effectiveness of two algorithms was verified through examples.

Keywords Rule base Merging of rule base Directed hypergraph Adjacency matrix Reachability matrix

0 引言

近年来,伴随着迅速发展的互联网和计算机技术,数据库数量和规模迅速增长,那么从诸多数据库中挖掘出来的相同或相似领域的知识并构建知识库的数量也越来越多。由于知识库的建立需要花费大量的人力和物力,因此利用已有的和相关的知识库进行合并从而获得新的知识库是必要的,并在此基础上进行行之有效的推理,这样可以降低目标问题在知识库中求解时间。

知识库管理系统是通过收集和总结相关领域知识、模型和专家经验建立独立的知识库,具有知识表示,模拟专家推理,结果获取和知识更新等功能。系统使用了基于规则的领域知识存储方式。本文通过已有的规则知识库融合生成新的规则知识库从而达到知识的重用与共享这一任务,可以改善新建知识库的效率。随着相同或相似领域的规则知识库数量逐渐增加,此方法变得尤为重要。大量知识库的合并将大大提高知识推理检索效率,但在带来便捷的同时,知识的冗余、冲突和环路是不可避免

的。针对上述问题,Revault^[1]提出模型转换技术可以有效地表达新规则;Adolfo^[2]合并两个本体,并考虑合并后本体的一致性、冲突和冗余的检测;Claes^[3]提出规则的合并方法并将其应用到历史数据处理中;文献[4]基于频繁模式算法提取规则并进行规则库的合并;文献[5-7]分别说明了规则库一致性和简洁性的检测,彭志平等^[8]合并多个知识库并将该技术应用到企业链中;Scholz等^[9]提出了一种检测冗余的算法,但是其算法较复杂,而且不适合规则较多的大知识库管理系统,也不能解决规则间环路和冲突的问题;安莉^[10]研究了规则知识库添加了新的规则后冗余规则的判别、处理以及校验的实现方法但并未对环路和冲突等问题进行研究;文献[11]研究了基于有向超图的规则库冗余及环路检测方法,该文献巧妙地利用超图表示规则库,并由超图相对应的可达矩阵检测规则的冗余和环路,而并未对规则的冲突提出检测的算法。

收稿日期:2014-07-23。国家自然科学基金项目(31171456,61203373);中国科学院战略性先导专项(XDA08040110)。贾秀芳,博士生,主研领域:粒计算,数据挖掘,专家系统。宋良图,研究员。王儒敬,研究员。李伟,助理研究员。

因此,为了避免规则知识库合并后出现规则间冗余、冲突和环路而影响推理效率以及推理准确性的不利后果,在文献[11]的基础上,本文提出一种用于合并规则知识库后规则库的一致性(是否存在规则冲突)和简洁性(是否存在冗余规则和环路)的检测方法。

本文首先将有向超图用于规则库的表示;其次构建其邻接矩阵,通过一种算法将两个邻接矩阵合并从而达到规则库合并;进而通过求可达矩阵判定规则库的冗余、环路和冲突;最后给出实例以表明规则库合并以及一致性和简洁性检测的可行性和有效性。

1 有向超图及其邻接矩阵

1.1 规则的表达形式

一个产生式规则的一般形式为 $P \rightarrow Q$,其左边 P 表示一组前提或状态,右边 Q 表示若干结论或动作,其含义是“如果前提 P 满足,那么可以推出结论 Q ”。其中,前提和结论都由谓词公式表示。为讨论方便,假定规则库的规则是确定性规则知识,同时假定规则库中的所有规则都是 Horn 子句,即每条规则的形式都满足以下两个条件:

- 1) 规则的前提是若干个文字的合取,且相同的文字在一条规则的前提中只能出现一次;
- 2) 规则的结论是一个文字。其中,一个文字是指一个原子公式或一个原子公式的否定。

事实上,任何一条规则都可以适当转换成等价的一条或若干条上述形式的规则。不失一般性,假设规则的格式为:

$$x_1 \wedge x_2 \wedge \dots \wedge x_n \rightarrow x_{n+1}$$

如果规则的前提只包含一个原子公式,则该规则称为简单规则,否则称为复合规则。

1.2 有向超图与邻接矩阵

有向超图是在无向超图中每条超边的基础上添加方向所得。一般认为,有向超图是由有限非空顶点集 V 和有向超边集 E 组成的有序对 $H = (V, E)$ 。有向超边 $e \in E$ 是有序对 (X, Y) 。这里的 X 和 Y 是 V 的不相交子集(可以为空),称 X 为有向超边 e 的尾点集,记为 $(T(e))$, Y 为有向超边 e 头点集,记为 $(H(e))$ 。在有向超图中,超边 e 所连接的是多个顶点组成的点集。这一性质适合于有向超图表示含有复合规则的规则库。

1.2.1 利用有向超图表示规则库

利用超图来表示规则库首先被 Valiente^[12] 提出。Nazareth 和 Kennedy^[13] 在 1991 年, Ramaswamy 等^[14] 在 1997 年提出使用有向图以清楚表示规则库中的规则,而不需要依赖人为地额外注释来表示这些关系。

利用超图来表示以下规则库 RB_1 , 如图 1 所示。

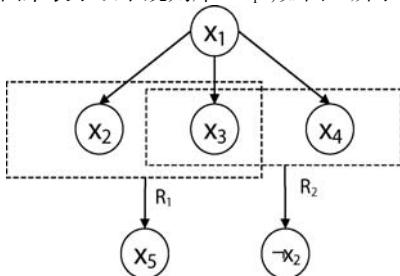


图 1 规则库 RB_1 的有向超图表示

- 规则 1: $x_1 \rightarrow x_2$;
- 规则 2: $x_1 \rightarrow x_3$;
- 规则 3: $x_1 \rightarrow x_4$;
- 规则 4: $x_2 \wedge x_3 \rightarrow x_5$, 记为 R_1 ;
- 规则 5: $x_3 \wedge x_4 \rightarrow \neg x_2$, 记为 R_2 。

1.2.2 有向超图的邻接矩阵

文献[11,15]中提出了超图的邻接矩阵,在文献[11]中主要研究的是规则的冗余和环路的检测而未对冲突的检测作进一步地研究。为了检测规则库合并后的冲突,本文在文献[11]的基础上重新给出有向超图相应的邻接矩阵的定义。

假设规则库中有 m 个原子公式,分别记为 x_1, x_2, \dots, x_m ; 并假设规则库中有 l 条复合规则,分别记为 R_1, R_2, \dots, R_l , 将规则库表示成有向超图。为了检测规则库合并后的冲突,本文定义的邻接矩阵是建立在原子公式和原子公式否定之上的。每个原子公式以及原子公式的否定构成邻接矩阵的行和列。因此邻接矩阵的规模取决于规则库中原子公式的数量,即如果规则库中有 m 个原子公式,则邻接矩阵的规模为 $2m \times 2m$ 。

定义有向超图的邻接矩阵 $A_{2m \times 2m}$ 为:

$$A_{ij} = \begin{cases} 0 & x_i \notin H(e) \text{ 且 } x_j = T(e), \forall e \in E \\ 1 & x_i = H(e) \text{ 且 } x_j = T(e), \forall e \in E \\ (x_i, R_k) & x_i \in H(e) \text{ 且 } x_j = T(e), \forall e \in E \end{cases}$$

即,如果规则库中不存在前提包含 x_i 且结论为 x_j 的规则,则记 $A_{ij} = 0$; 如果规则库中存在简单规则前提为 x_i 且结论为 x_j 的规则,则记 $A_{ij} = 1$; 如果规则库中存在复合规则 R_k , 前提包含 x_i 且结论为 x_j , 则 A_{ij} 表示为 (x_i, R_k) 。

根据邻接矩阵的定义可得有向超图 1 的邻接矩阵为:

$$A = \begin{bmatrix} 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & (x_2, R_1) & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & (x_3, R_1) & 0 & (x_3, R_2) & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & (x_4, R_2) & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

1.3 邻接矩阵合并的算法

一般情况下需要合并的两个规则库规模不同,假定两个规则库分别有 m_1, m_2 个原子公式,两个规则库的邻接矩阵分别为 $A = (a_{ij})_{2m_1 \times 2m_1}, B = (b_{ij})_{2m_2 \times 2m_2}$ 。两规则库合并后形成的规则库的规模是 $m = \max\{m_1, m_2\}$ 。

若 $m_1 < m_2$, 则令:

$$a_{i, m_1+1} = a_{i, m_1+2} = \dots = a_{i, m_2} = a_{i, 2m_1+1} = a_{i, 2m_1+2} = \dots = a_{i, 2m_2} = 0$$

$$a_{m_1+1, j} = a_{m_1+2, j} = \dots = a_{m_2, j} = a_{2m_1+1, j} = a_{2m_1+2, j} = \dots = a_{2m_2, j} = 0$$

若 $m_1 > m_2$:

$$a_{i, m_2+1} = a_{i, m_2+2} = \dots = a_{i, m_1} = a_{i, 2m_2+1} = a_{i, 2m_2+2} = \dots = a_{i, 2m_1} = 0$$

$$a_{m_2+1, j} = a_{m_2+2, j} = \dots = a_{m_1, j} = a_{2m_2+1, j} = a_{2m_2+2, j} = \dots = a_{2m_1, j} = 0$$

A, B 分别为两超图对应的邻接矩阵,那么邻接矩阵 A 与邻接矩阵 B 合并为邻接矩阵 $C = \{c_{ij}\}_{2m \times 2m}$,其中求 c_{ij} 的算法如下:

输入:

$$A = \{a_{ij}\}_{2m_1 \times 2m_1}, B = \{b_{ij}\}_{2m_2 \times 2m_2}$$

输出:

$$C = \{c_{ij}\}_{2m \times 2m}, \text{其中 } m = \max\{m_1, m_2\}$$

步骤:

```

for(i = 1, i ≤ 2m, i++)
{
for(j = 1, j ≤ 2m, j++)
{
if( aij = bij = 0 )
    cij = 0 ;
if( aij = 0 and bij ≠ 0 )
    cij = bij ;
if( aij ≠ 0 and bij = 0 )
    cij = aij ;
if ( aij = 1 and bij ≠ 0 )
    cij = 1 ;
if ( aij ≠ 0 and bij = 1 )
    cij = 1 ;
if (
    aij = (xi, R) and bij = (xi, R')
)
    cij = (xi, R) ∪ (xi, R')
}
}
    
```

1.4 超图到可达矩阵的转变

1.4.1 可达矩阵的定义

定义 1 设有向超图 $H = (V, E)$

$$V = \{v_1, v_2, \dots, \neg v_1, \neg v_2, \dots, \neg v_m\}$$

$$= \{x_1, x_2, \dots, x_{m+1}, \dots, x_{2m}\}$$

令:

$$p_{ij} = \begin{cases} 0 & \text{从 } x_i \text{ 到 } x_j \text{ 不存在任何通路} \\ 1 & \text{从 } x_i \text{ 到 } x_j \text{ 存在通路, } 0 \leq i, j \leq 2m \end{cases}$$

则称矩阵 $P(D) = (p_{ij})_{2m \times 2m}$ 为有向超图 D 的可达性矩阵。

k 步可达矩阵 D^k 和总可达矩阵 E^k 的定义源于文献[11]。

定义 2 k 步可达矩阵 D^k

$D^k_{i,j} = a$; 从 x_i 到 x_j 存在 a 条长度为 k 的路径。

定义 3 总可达矩阵 E^k

$E^k_{i,j} = b$; 从 x_i 到 x_j 存在 b 条长度不大于 k 的路径。

$$\text{显然 } E^k = \sum_{i=1}^k D^i$$

观察总可达矩阵 E^k :

$E^k_{i,j} = 0$ 说明 x_i 无法到达 x_j ;

$E^k_{i,j} = 1$ 说明 x_i 可达 x_j 且只有一条路径;

$E^k_{i,j} \geq 2$ 说明 x_i 到 x_j 存在多条路径,即规则库存在冗余;

$E^k_{i,i} \geq 1$ 则规则库存在环路;

$E^k_{i,j} = 1$ 且 $E^k_{i,(j+m)} = 1$, 则规则库存在规则冲突。

为了便于后续的计算,由超图可得路径长度的矩阵 $F_{n \times n} = (F_{i,j})_{2m \times 2m}$ 。其中 $F_{i,j} = c$, c 为 x_i 到 x_j 的最长路径长度,且从 x_i 到 x_j 只遍历一次。例如,图 2 中所示 x_1 到 x_4 的最长路径长度为 2, x_1 到 x_2 的最长路径长度为 1。

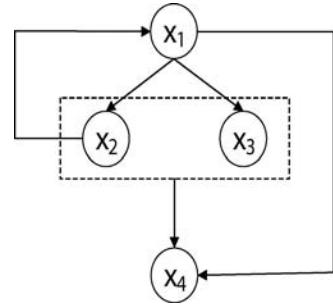


图 2 有向超图的路径长度

1.4.2 计算 E^{k+1}

从邻接矩阵的定义可知,将 $A_{2m \times 2m}$ 中非 1 的元素赋值为 0, 即得到 $D^1_{2m \times 2m}$ 与总可达矩阵 $E^1_{2m \times 2m}$ 。

假设 D^k 和 E^k 均已求得,欲求 E^{k+1} , 需要计算 D^{k+1} 。

那么:

$$D^{k+1} = D^k \cdot A$$

$$= \begin{bmatrix} D^k_{1,1} & D^k_{1,2} & \dots & D^k_{1,m} & D^k_{1,m+1} & \dots & D^k_{1,2m} \\ D^k_{2,1} & D^k_{2,2} & \dots & D^k_{2,m} & D^k_{2,m+1} & \dots & D^k_{2,2m} \\ \vdots & \vdots & & \vdots & \vdots & & \vdots \\ D^k_{2m,1} & D^k_{2m,2} & \dots & D^k_{2m,m} & D^k_{2m,m+1} & \dots & D^k_{2m,2m} \end{bmatrix} \cdot \begin{bmatrix} A_{1,1} & A_{1,2} & \dots & A_{1,m} & A_{1,m+1} & \dots & A_{1,2m} \\ A_{2,1} & A_{2,2} & \dots & A_{2,m} & A_{2,m+1} & \dots & A_{2,2m} \\ \vdots & \vdots & & \vdots & \vdots & & \vdots \\ A_{2m,1} & A_{2m,2} & \dots & A_{2m,m} & A_{2m,m+1} & \dots & A_{2m,2m} \end{bmatrix} \quad (1)$$

$$\text{其中 } D^{k+1}_{i,j} = \sum_{l=1}^{2m} D^k_{i,l} A_{l,j}$$

规定:

若 $D^k_{i,l} = 0$ 或 $A_{l,j} = 0, l = 1, 2, \dots, 2m$, 则 $D^{k+1}_{i,j} = 0$ 。

若 $D^k_{i,j} = 1 \cdot (x_p, R') = (x_p, R')$, 且记 $R': x_p \wedge x_q \rightarrow x_j$, 则:

$$D^{k+1}_{i,j} = \begin{cases} 1 & F_{i,q} \leq k \\ 0 & \text{其他} \end{cases} \quad (2)$$

若 $D^k_{i,j} = (x_p, R') + (x_q, R')$, 且记 $R': x_p \wedge x_q \rightarrow x_j$, 则:

$$D^{k+1}_{i,j} = \begin{cases} 1 & F_{i,p} = k, F_{i,q} \leq k \text{ 或 } F_{i,q} = k, F_{i,p} \leq k \\ 0 & \text{其他} \end{cases} \quad (3)$$

2 规则冗余、环路和冲突检测的算法

规则库合并后规则冗余、环路和冲突检测的算法流程:

(1) 计算邻接矩阵 $A_{n \times n}$, 路径长度矩阵 $F_{n \times n} = (F_{i,j})_{n \times n}$, 一步可达矩阵 $D^1_{n \times n}$, 一步总可达矩阵 $E^1_{n \times n}$ 。如果检测到环路, 即 $E^1_{i,i} \geq 1$, 则令 $D^1_{i,i} = 0$, 并执行(2), 否则, 直接转入(2);

(2) 计算并判断 $D^k_{n \times n} = D^{k-1}_{n \times n} \cdot A$

If $D^k_{n \times n} = 0$

Then 终止循环, 执行(3)

Else 执行(4)

(3) 计算 $E^k_{n \times n} = E^{k-1}_{n \times n} + D^k_{n \times n} = E^{k-1}_{n \times n}$

If $E^k_{i,j} \geq 2$ Then 规则库中的规则存在冗余;

If $E^k_{i,i} \geq 1$ Then 规则库中的规则存在环路;

If $E^k_{i,j} = 1$ and $E^k_{i,(j+m)} = 1$

Then 规则库中规则存在冲突;

(4) 计算 $E^{k+1}_{n \times n} = E^k_{n \times n} + D^k_{n \times n}$

If $E^{k+1}_{i,i} \geq 1$

Then 执行(5)

Else 令 $k = k + 1$, 执行(2)

(5) 令 $D_{i,i}^k = 0, k = k + 1$, 并执行(2)。

3 实例

中知识管理库系统里知识融合模块的功能是对两个同领域的规则知识库进行融合, 并对融合后的规则进行冗余、冲突和环路的检测, 进而形成新的规则知识库, 达到知识的重用和共享, 以此提高建立知识库的效率。

两个算法具体实现方法通过举例说明:

RB_1 是专家 a 建立的水稻病虫害规则知识库中的部分规则。

RB_2 是专家 b 建立的水稻病虫害规则知识库中的部分规则。

以下是 RB_2 中的 8 条规则:

规则 1: $x_1 \rightarrow x_2$;

规则 2: $x_1 \rightarrow x_3$;

规则 3: $x_1 \rightarrow x_4$;

规则 4: $x_1 \rightarrow x_5$;

规则 5: $x_2 \wedge x_3 \rightarrow x_5$, 记为 R'_1 ;

规则 6: $x_2 \wedge x_3 \rightarrow x_6$, 记为 R'_2 ;

规则 7: $x_4 \rightarrow \neg x_6$;

规则 8: $x_5 \wedge x_6 \rightarrow x_1$, 记为 R'_3 。

以规则 5 为例“若体色灰黄和前后翅外缘有暗灰色宽带纹, 那么诊断结果: 水稻二化螟”。

利用有向超图表示规则库 RB_2 , 如图 3 所示。

$$C = \begin{bmatrix} 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & (x_2, R_1) \cup (x_2, R'_1) & (x_2, R'_2) & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & (x_3, R_1) \cup (x_3, R'_1) & (x_3, R'_2) & 0 & (x_3, R_2) & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & (x_4, R_2) & 0 & 0 & 0 & 1 \\ (x_5, R'_3) & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ (x_6, R'_3) & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

邻接矩阵 C 所对应的有向超图如图 4 所示。

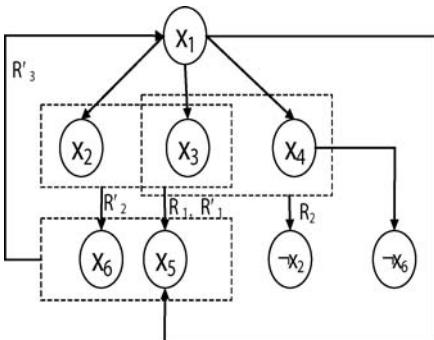


图 4 合并规则库的有向超图表示

由有向超图 4 得到路径长度矩阵为:

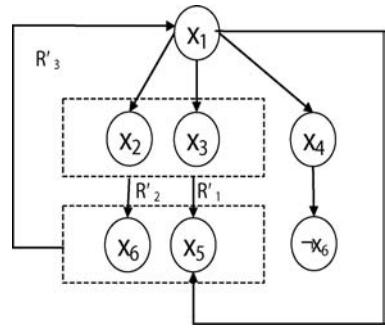


图 3 规则库 RB_2 的有向超图表示

根据有向超图邻接矩阵的定义可知, RB_2 所对应应有向超图的邻接矩阵为:

$$B = \begin{bmatrix} 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & (x_2, R'_1) & (x_2, R'_2) & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & (x_3, R'_1) & (x_3, R'_2) & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ (x_5, R'_3) & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ (x_6, R'_3) & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

根据邻接矩阵合并的算法可知邻接矩阵 A 和邻接矩阵 B 合并后的邻接矩阵 C 为:

$$C = \begin{bmatrix} 3 & 1 & 1 & 1 & 2 & 2 & 0 & 2 & 0 & 0 & 0 & 2 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

邻接矩阵 C 的一步可达矩阵分别为:

$$D_{12 \times 12}^1 = E_{12 \times 12}^1 = \begin{bmatrix} 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

检测 $E_{12 \times 12}^1$, 未检测到环路;
根据式(1)可得:

$$D_{12 \times 12}^2 = D_{12 \times 12}^1 \cdot C$$

其中:

$$D_{1,1}^2 = 1 \cdot (x_5, R'_3) = (x_5, R'_3)$$

存在规则 $R'_3: x_5 \wedge x_6 \rightarrow x_1$, 但是 $F_{1,6} = 2 > 1$, 根据式(2)可知 $D_{1,1}^2 = 0$ 。

$$D_{1,5}^2 = (x_2, R_1) \cup (x_2, R'_1) + (x_3, R_1) \cup (x_3, R'_1)$$

存在规则 $R_1: x_2 \wedge x_3 \rightarrow x_5, R'_1: x_2 \wedge x_3 \rightarrow x_5$ 由于 $F_{1,2} = F_{1,3} = 1$, 根据式(3)可知:

$$D_{1,5}^2 = \{(x_2, R_1) + (x_3, R_1)\} \cup \{(x_2, R'_1) + (x_3, R'_1)\} = 2$$

$$D_{1,6}^2 = (x_2, R'_2) + (x_3, R'_2)$$

存在规则 $R'_2: x_2 \wedge x_3 \rightarrow x_6$, 由于 $F_{1,2} = F_{1,3} = 1$, 根据式(3)可知 $D_{1,6}^2 = 1$ 。

$$D_{1,8}^2 = (x_3, R_2) + (x_4, R_2)$$

存在规则 $R_2: x_3 \wedge x_4 \rightarrow \neg x_2$, 由于 $F_{1,3} = F_{1,4} = 1$, 根据式(3)可知 $D_{1,8}^2 = 1$ 。

因此:

$$D_{12 \times 12}^2 = \begin{bmatrix} 0 & 0 & 0 & 0 & 2 & 1 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$E_{12 \times 12}^2 = D_{12 \times 12}^1 + D_{12 \times 12}^2 = \begin{bmatrix} 0 & 1 & 1 & 1 & 3 & 1 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

检测 $E_{12 \times 12}^2$, 未检测到环路;
同理:

$$D_{12 \times 12}^3 = D_{12 \times 12}^2 \cdot C = \begin{bmatrix} 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$E_{12 \times 12}^3 = D_{12 \times 12}^1 + D_{12 \times 12}^2 + D_{12 \times 12}^3$$

$$= \begin{bmatrix} 2 & 1 & 1 & 1 & 3 & 1 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

根据第 2 节规则冗余、环路和冲突检测算法的步骤(4)和步骤(5)可知,若 $E_{1,1}^3 = 2 > 1$,那么令 $D_{1,1}^3 = 0$,此时 $D_{12 \times 12}^3 = 0$ 。再根据算法步骤(2)可知 $D_{12 \times 12}^4 = 0$,所以执行步骤(3)的结果为 $E_{1,1}^3 = 2$,检测到环路和冗余,即 x_1 可达 x_1 而且可以通过两条路径可达; $E_{1,5}^3 = 3$,检测到冗余,即 x_1 可以通过三条路径可达 x_5 ; $E_{1,6}^3 = 1$ 且 $E_{1,12}^3 = E_{1,6+6}^3 = 1$,检测到冲突,即 $x_1 \rightarrow x_6, x_1 \rightarrow \neg x_6$ 。

4 结 语

本文提出了一种用于知识库管理系统规则库的合并算法,此方法降低了系统的复杂性,大大提高了知识推理检索效率。同时利用可达矩阵检测了合并后规则的冗余、环路和冲突。通过实例验证了两个算法的有效性和可行性。规则库中的规则是不确定性规则知识时,规则库合并以及对合并后不确定性规则冗余、环路和冲突的检测有待于作进一步的研究。

参 考 文 献

[1] Revault N. Model transformation based on production rules[J]. Electronic Notes in Theoretical Computer Science, 2003, 72(4) : 1-14.
[2] Adolfo G A, Cuevas A D. Knowledge Accumulation through Automatic Merging of Ontologies[J]. Expert Systems with Applications, 2010, 37(3) : 1991-2005.

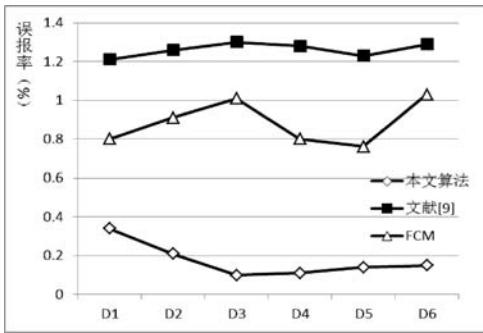


图7 算法误报率对比

图6、图7为不同算法对各测试集的检测率和误报率的对比,从对比中可以看出,本文提出的FCM-C4.5双过滤检测机制在KDDCup99数据集上的测试效果明显优于传统的FCM算法和文献[9]改进的聚类算法。

4 结语

本文提出了一种基于FCM-C4.5的双过滤入侵检测机制,该机制首先利用FCM算法将明显的正常数据过滤掉,然后利用C4.5算法对剩下的正常数据和异常数据进行再次过滤。通过KDD CUP 99数据集实验表明,该机制充分发挥了FCM能检测到未知攻击的能力与C4.5低误报率和对已知攻击高检测率的优点,既克服了FCM检测率低的问题,又解决了C4.5对未知攻击检测能力差的问题;同时通过逐层过滤的方式减少了第二层需要过滤的数据量,达到效率与精度的共同提高。本文参数较多,实验中参数阈值是通过多次试验选取的,下一步研究是寻找一个自适应的方法来选取最优参数。

参考文献

- [1] Fisch D, Hofmann A, Sick B. On the versatility of radial basis function neural networks: A case study in the field of intrusion detection[J]. Information Sciences, 2010, 180(12): 2421-2439.
- [2] Wang L, Jajodia S, Singhal A, et al. k-Zero day safety: A network security metric for measuring the risk of unknown vulnerabilities[J]. IEEE Transactions on Dependable and Secure Computing, 2014, 11(1): 30-44.
- [3] Li Y, Xia J, Zhang S, et al. An efficient intrusion detection system based on support vector machines and gradually feature removal method[J]. Expert Systems with Applications, 2012, 39(1): 424-430.
- [4] Wu S X, Banzhaf W. The use of computational intelligence in intrusion detection systems: A review[J]. Applied Soft Computing, 2010, 10(1): 1-35.
- [5] Guo C, Zhou Y J, Ping Y, et al. Efficient intrusion detection using representative instances[J]. Computers & Security, 2013, 39(Part B): 255-267.
- [6] Kim G, Lee S, Kim S. A novel hybrid intrusion detection method integrating anomaly detection with misuse detection[J]. Expert Systems with Applications, 2014, 41(4): 1690-1700.
- [7] Park N H, Oh S H, Lee W S. Anomaly intrusion detection by clustering transactional audit streams in a host computer[J]. Information Sciences, 2010, 180(12): 2375-2389.
- [8] Wang G, Hao J, Ma J, et al. A new approach to intrusion detection using Artificial Neural Networks and fuzzy clustering[J]. Expert Systems with Applications, 2010, 37(9): 6225-6232.
- [9] 杜强, 孙敏. 基于改进聚类分析算法的入侵检测系统研究[J]. 计

算机工程与应用, 2011, 47(11): 106-108.

- [10] SivathaSindhu S S, Geetha S, Kannan A. Decision tree based light weight intrusion detection using a wrapper approach[J]. Expert Systems with Applications, 2012, 39(1): 129-141.
 - [11] 蔡洪民, 伍乃骥, 滕少华. 分布式木马检测系统设计实现[J]. 计算机应用与软件, 2012, 29(5): 278-280.
 - [12] Sengupta N, Sen J, Sil J, et al. Designing of on line intrusion detection system using rough set theory and Q-learning algorithm[J]. Neurocomputing, 2013, 111(2): 161-168.
 - [13] Lee S, Kim G, Kim S. Self-adaptive and dynamic clustering for online anomaly detection[J]. Expert Systems with Applications, 2011, 38(12): 14891-14898.
 - [14] Lei J Z, Ghorbani A A. Improved competitive learning neural networks for network intrusion and fraud detection[J]. Neurocomputing, 2012, 75(1): 135-145.
 - [15] 陈焯, 刘渊. 基于参数优化SVM融合的网络异常检测[J]. 计算机应用与软件, 2013, 30(9): 39-43.
 - [16] Teng S H, Du H L, Wu N Q, et al. A Cooperative Network Intrusion Detection Based on Fuzzy SVMs[J]. Journal of Networks, 2010, 5(4): 475-483.
 - [17] Stolfo S J, Fan W, Lee W, et al. KDD CUP 1999 data [DB/OL]. [1999]. <http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html>.
 - [18] 肖立中, 邵志清, 马汉华, 等. 网络入侵检测中的自动决定聚类数算法[J]. 软件学报, 2008, 19(8): 2140-2148.
- ~~~~~
- (上接第253页)
- [3] Claes J, Poels G. Merging Event Logs for Process Mining: A Rule Based Merging Method and Rule Suggestion Algorithm[J]. Expert Systems with Applications, 2014, 41(16): 7291-7306.
 - [4] 王文娟, 李炳龙. IDS规则库构建与合并算法[J]. 计算机应用与软件, 2010, 27(11): 259-261.
 - [5] 徐珊珊, 董利达, 朱丹, 等. 一类活性Petri网控制器的冗余检测及结构简化[J]. 控制理论与应用, 2013, 30(6): 673-682.
 - [6] 江先伟, 王军祥. 规则不一致消解网格服务的设计与实现[J]. 电子科技, 2013, 26(12): 138-140.
 - [7] 倪俊, 陈晓苏, 刘辉宇, 等. 网络安全策略求精一致性检测盒冲突消解机制的研究[J]. 计算机科学, 2011, 38(2): 32-37.
 - [8] 彭志平, 夏战锋, 周超. 多知识库整合技术在企业链中的应用[J]. 计算机工程, 2012, 38(2): 82-84.
 - [9] Schmolze J G, Snyder W. Detecting Redundancy Among Production Rules Using Term Rewrite Semantics[J]. Knowledge-based System, 1999, 12(1-2): 3-11.
 - [10] 安莉, 王建林. 用于专家系统规则库的冗余校验方法研究[J]. 计算机工程与应用, 2008, 44(34): 191-193.
 - [11] 孙伟, 郭莉, 高天一, 等. 一种基于有向超图的规则库冗余及环路检测方法[J]. 大连理工大学学报, 2008, 48(1): 74-78.
 - [12] Valiente G. Verification of Knowledge based on Redundancy and Subsumption Using Transformations[J]. International Journal Expert Systems, 1993, 6(3): 341-355.
 - [13] Nazareth D L, Kennedy M H. Verification of Rule-Based Knowledge Using Directed Graphs[J]. Knowledge Acquisition, 1991, 3(4): 339-360.
 - [14] Ramaswamy M, Sarkar S, Chen Y S. Using Directed Hypergraphs to Verify Rule-Based Expert Systems[J]. Transactions on Knowledge and Data Engineering, 1997, 9(2): 221-237.
 - [15] Nuffelen C V. On Adjacency Matrices for Hypergraphs[J]. Annals of Discrete Mathematics, 1980(9): 181.