6 第六章　关系数据理论

Principles of Database Systems

第六章 关系数据理论

6.3、Armstrong公理系统

1. 逻辑蕴涵，闭包
2. Armstrong公理系统（3+3）
3. 属性闭包
4. 等价、覆盖和最小函数依赖集

问题：

- 对于给定的一组函数依赖，如何判断另外的一些函数依赖是否成立?
- 如何找出R上所有的函数依赖?
- 码如何求?

——一套有效而完备的公理推理系统——Armstrong公理系统。

6.3 数据依赖的公理系统

定义6.11（逻辑蕴含）对于关系模式R<U，F>，其任何一个关系r，若函数依赖X → Y都成立（即r中任意两元组s，t，若t[X]=s[X]，则t[Y]=s[Y]），则称**函数依赖集F逻辑蕴含X→Y，或X→Y从F推导出来的，或X→Y 逻辑蕴含于F** 。

定义6.12 在关系模式R（U，F）中为F所逻辑蕴含的函数依赖的**全体**叫做**F的闭包**，记作F^+ 。

问题：

1）如何从已知的F出发，推出F^+中的所有函数依赖?

2）已知F和X、Y，如何判断X→Y是否在F^+中?

根据已知的F出发推导出新的函数依赖，需要使用一些推理规则。1974年，W.W.Armstrong总结了各种规则，形成了著名的Armstrong公理系统。

6.3 数据依赖的公理系统

Armstrong公理的内容:

设有关系模式R(U, F)，U为属性全集，F是U上的函数依赖集，X，Y，Z $\subseteq$ U。则有:

- **自反律：** 若Y $\subseteq$ X $\subseteq$ U，则X －＞ Y为F所蕴含(给出平凡的函数依赖)。
- **增广律：** 若X → Y为F所蕴含，且Z $\subseteq$ U，则XZ →YZ为F所蕴含。
- **传递律：** 如X → Y及Y → Z为F 所蕴含，则X → Z为F所蕴含。

Armstrong公理的正确性

- 设r是R(U, F)上的任一关系，t，s $\in$ r
- **自反律：** 若Y $\subseteq$ X， 则X→Y

$$\left.\begin{matrix} t[X] = s[X] \\ Y \subseteq X \end{matrix}\right\} \Rightarrow t[Y] = s[Y] \Rightarrow X \to Y$$

- **增广律：** 若X→Y，则XZ→YZ

$$t[XZ] = s[XZ] \Rightarrow \left.\begin{matrix} t[X] = s[X] \\ X \to Y \end{matrix}\right\} \Rightarrow t[Y] = s[Y]$$

$$t[XZ] = s[XZ] \Rightarrow t[Z] = s[Z]$$

$$\left.\begin{matrix} t[Y] = s[Y] \\ t[Z] = s[Z] \end{matrix}\right\} \Rightarrow t[YZ] = s[YZ] \Rightarrow XZ \to YZ$$

Armstrong公理的正确性

- **传递律：若X→Y，Y→Z，则X→Z**

$$\left.\begin{array}{c} X \to Y \\ t[X] = s[X] \end{array}\right\} \Longrightarrow \left.\begin{array}{c} t[Y] = s[Y] \\ Y \to Z \end{array}\right\} \Longrightarrow t[Z] = s[Z]$$

$$\Longrightarrow X \to Z$$

- **Armstrong公理的推论**

❖ **合成规则：若X→Y，X→Z，则X→YZ**

❖ **分解规则：若X→YZ，则X→Y，X→Z**

❖ **伪传递规则：若X→Y，YW→Z，则XW→Z**

- **思考：如何使用Armstrong公理证明推论的正确性?**

Armstrong公理推论的证明

1) 合成规则（若X→Y，X→Z，则X→YZ)

∵X→Z，∴有XX→XZ，即X→XZ （增广律）

又∵X→Y，∴ XZ→YZ （增广律）

∴X→YZ（传递律）

2) 分解规则（若X→YZ，则X→Y，X→Z)

∵Y ⊆YZ ⊆ U，∴YZ→Y（自反律）

同理YZ→Z（自反律）

∵X→YZ，∴X→Y（传递律）

同理X→Z（传递律）

3) 伪传递规则（若X→Y，YW→Z，则XW→Z)

∵X→Y，∴XW→YW（增广律，两边扩充W）

∵YW→Z，∴XW→Z（传递律）

> 引理6.1：如果A_i（i=1,2,3,…,n）是关系模式R的属性，则$X \rightarrow A_1A_2 \ldots A_n$的充要条件是$X \rightarrow A_i$（i=1,2,…,n）均成立。

课堂练习

- **课堂练习：** 已知关系模式R(U, F)，U = (A, B, C, G, H, I)，F = {A→B, A→C, CG→H, CG→I, B→H}，判断下列函数依赖是否为F的逻辑蕴涵?

- A→ H 是
- CG → HI 是
- AG → I 是

问题：能不能用一种一般性的算法来判定X→Y是否是F的逻辑蕴涵?

F的闭包

- **例如：从F = {X→A$_1$，X→A$_2$，…，X→A$_n$}出发可推导出2^{n}个不同的函数依赖。**

F={X→Y, Y→Z}

F^+={

X→φ,	Y→φ,	Z→φ,	XY→φ,	XZ→φ,	YZ→φ,	XYZ→φ,
X→X,	Y→Y,	Z→Z,	XY→X,	XZ→X,	YZ→Y,	XYZ→X,
X→Y,	Y →Z,		XY→Y,	XZ→Y,	YZ→Z,	XYZ→Y,
X→Z,	Y→YZ,		XY→Z,	XZ→Z,	YZ→YZ,	XYZ→Z,
X→XY,			XY→XY,	XZ→XY,		XYZ→XY,
X→XZ,			XY→YZ,	XZ→XZ,		XYZ→YZ,
X→YZ,			XY→XZ,	XZ→XY,		XYZ→XZ,
X→XYZ,			XY→XYZ,	XZ→XYZ,		XYZ→XYZ }

F={X→A1, …… , X→An}的闭包F^+计算是一个NP完全问题

属性闭包

- **定义6.13：** 设有关系模式R(U, F)，U = {A_1，A_2，…，A_n}，X⊆U，F是U上的一个函数依赖集，则称所有用Armstrong公理从F推导出的函数依赖$X \rightarrow A_i$中所有A_i的属性集合为**属性集X关于F的闭包**，记为X_F^+。即：
    - X_F^+ = { A | X→A能由F根据Armstrong公理推导出 }

- **例：** 在关系模式R(U, F)中，U={A, B, C}，F={A→B, B→C}，则A、B、C关于F的闭包为：

  $A_F^+ = ABC$，
  $B_F^+ = BC$，
  $C_F^+ = C$

属性闭包

引理6.2：函数依赖X→Y能由F根据Armstrong公理推导出来的充要条件是 $Y \subseteq X_F^+$。（证明略）

由该定理可知，判定X→Y是否能由F根据Armstrong公理导出，可转化为：

求X_F^+，判定$Y \subseteq X_F^+$是否成立。

计算属性闭包的算法

输入： X，F

输出： X_F^+

方法：

(1) $X^{(0)} = \phi$，$X^{(1)} = X$；

(2) 如果$X^{(0)} \neq X^{(1)}$，置$X^{(0)} = X^{(1)}$，否则转（4）；

(3) 对F中的每个函数依赖Y→Z，若$Y \subseteq X^{(0)}$，置 $X^{(1)} = X^{(1)} \cup Z$，即将Y的右部并入$X^{(1)}$中。转（2）；

(4) 输出$X^{(1)}$，即为X_F^+。

属性闭包计算举例

- **例：** 设有关系模式R(U, F)，U= (A, B, C, D, E), F = {AB→C, B→D, C→E, EC→B, AC→B}，计算$(AB)_F^+$ 。

	所用依赖	$X^{(0)}$	$X^{(1)}$
初始值		ϕ	AB
第一遍	AB→C, B→D	AB	ABCD
第二遍	C→E, AC→B	ABCD	ABCDE
第三遍	EC→B	ABCDE	ABCDE

计算结果： $(AB)_F^+$ = ABCDE

通过计算属性闭包可以判断一个属性组是否为关系的码。

如本例中：∵ $(AB)_F^+$ = ABCDE =U, 且$A_F^+ \neq$U， $B_F^+ \neq$U

∴AB为R的一个码。

课堂练习

- **设关系模式R(B，O，I，S，Q，D)，函数依赖集F={S→D，I→S，IS→Q，B→Q，S→O，D→I}。找出R的所有候选码，并指出R最高属于第几范式。**

输入：关系模式R的属性集U，及其函数依赖集F
输出：R的所有候选码集合K
步骤：
（1）令K＝ϕ;
（2）求从未在F中函数依赖的右部出现过的属性集X;
（3）求X_F^+，若X_F^+＝U，则令K＝{X}，转（7）；
（4）求在F中函数依赖左右部都出现过的属性集Y;
（5）依次取Y中每个属性 (设为A)，求$(XA)_F^+$，若$(XA)_F^+$＝U， 则令K＝K ∪ {XA};
（6）依次取Y中每两个、三个…（设为Z），若XZ不包含K中的任一候选码，则求$(XZ)_F^+$，若$(XZ)_F^+$＝U，则令K＝K∪ {XZ};
（7）输出K中所有候选码。

Armstrong公理

定理：Armstrong公理是有效的、完备的。

- **有效性：** 由F出发，根据Armstrong公理推导出来的每一个函数依赖一定在F^+中。
- **完备性：** F^+中的每一个函数依赖，必定可以由F出发，根据Armstrong公理推导出来。
- 证明：

1. 有效性：可由引理6.1得证
2. 完备性：只需证明**逆否命题**：若函数依赖$X \rightarrow Y$不能由F从Armstrong公理导出，那么它必然不为F所蕴含。

> **引理6.1：** 如果A_i（i=1,2,3,…,n）是关系模式R的属性，则$X \rightarrow A_1A_2…A_n$的**充要条件**是$X \rightarrow A_i$（i=1,2,…,n）均成立。

Armstrong公理完备性证明

即：存在一个关系r，F中所有的函数依赖都满足r，而不能用公理推出的X→Y却不满足r，即F不能逻辑蕴涵X→Y。证明就是要找到这个r。

证明:

(1) 引理: 若$V\rightarrow W$成立，且$V\subseteq X_F^+$，则$W\subseteq X_F^+$

(2) 构造一张二维表r，它由下列两个元组构成，可以证明r必是$R(U,F)$的一个关系，即F^+中的全部函数依赖在 r上成立。

X_F^+	$U-X_F^+$
11......1	00......0
11......1	11......1

如果一个属性集不完全属于X_F^+，则该属性集在2个元组上的属性值必不相等。

(3) 若$X\rightarrow Y$不能由F从Armstrong公理导出，则Y不是X_F^+的子集。

$X\subseteq X_F^+$，而$Y\nsubseteq X_F^+$，X在r上2个元组上一致，而Y在r的2个元组上不一致。

函数依赖的等价和覆盖

❖ **定义6.14**

- **等价**：若F和G是R的两个函数依赖集，如果$F^+ = G^+$, 则称F等价于G。
- **覆盖**：若F和G是R的两个等价的函数依赖集$F^+ = G^+$，则称F覆盖G，同时G也覆盖F。

> **引理6.3**：设F和G是R的两个函数依赖集，则F和G等价的充分必要条件是$F \subseteq G^+$ 且 $G \subseteq F^+$。

证明:

1）必要性

若$F^+ = G^+$，显然有$F \subseteq F^+ \subseteq G^+$，$G \subseteq G^+ \subseteq F^+$。

2）充分性

若$F \subseteq G^+$，$G \subseteq F^+$

则 $F^+ \subseteq (G^+)^+$，即$F^+ \subseteq G^+$；$G^+ \subseteq (F^+)^+$，即$G^+ \subseteq F^+$

$\therefore G^+ = F^+$。 (证毕)

最小函数依赖集

华中科技大学 Huazhong University Of Science & Technology

定义6.15 最小函数依赖集

若函数依赖F满足下列条件，则称F为一个最小函数依赖集，记为F_m。

1）F中每个函数依赖的**右**部都是**单属性**；

2）对于F中的任一函数依赖X→A，F-{X→A}与F是不等价的（**即F中不存在多余的依赖**）

3）对于F中的任一函数依赖X→A，不存在X的子集Z，使得F与(F-{X→A})∪{Z→A} 等价（**即左部无多余属性**）

F的最小依赖集求解算法:

1）用**分解**规则将F中所有函数依赖的右部分解为单属性的函数依赖，去掉重复依赖；

2）**去掉多余依赖：**对每个依赖X→Y，令G = F - {X→Y}，求X_G^+，若Y ⊆ X_G^+，则X→Y为多余依赖，将其从F中去掉；

3）**去掉依赖左部的多余属性：**对每个左部为多属性的依赖，如X→A，设X = $B_1B_2...B_m$，逐一考察B_i，若A∈$(X - B_i)_F^+$，则B_i是多余属性，用X - B_i代替X。

重复23，直到Fm不再改变

最小函数依赖集计算举例

例： 已知F = { AB→C, C→A, BC→D, ACD→B, D→EG, BE→C, CG→BD, CE→AG }，求F的最小依赖集F_m。

解：

1）**将F中所有函数依赖右部不为单属性的化为单属性**

F_1 = { AB→C, C→A, BC→D, ACD→B, D→E, D→G, BE→C, CG→B, CG→D, CE→A, CE→G }

2）**去掉F_1中多余的函数依赖**

对AB→C，令G = F_1-{AB→C}，计算$(AB)_G^+$ =AB

∵C⊄$(AB)_G^+$，∴AB→C不能去掉；

对C→A，令G = F_1-{C→A}，计算C_G^+ =C

∵A⊄C^+，∴C→A不能去掉；

……

F_2 = {AB→C, C→A, BC→D, D→E, D→G, BE→C, CG→B, CE→G }

最小函数依赖集计算举例

F_2 = {AB→C, C→A, BC→D, D→E, D→G, BE→C, CG→B, CE→G }

3）**去掉F_2中函数依赖左部多余属性**

对AB→C，在F_2中分别计算：

对A，求$B_{F2}^+ = B$，因为$C \not\subset B_{F2}^+$，所以A不是多余属性。

对B，求$A_{F2}^+ = A$，因为$C \not\subset A_{F2}^+$，所以B不是多余属性。

对BC→D，在 F2中分别计算：

对B，求$C_{F2}^+ = CA$，因为$D \not\subset C_{F2}^+$，所以B不是多余属性。

对C，求$B_{F2}^+ = B$，因为$D \not\subset B_{F2}^+$，所以C不是多余属性。

……

$\because F_2$中函数依赖左部无多余属性， $\therefore F_3 = F_2$

$\therefore F_m = F_2$

注意：F的最小函数依赖集不是唯一的，与计算顺序有关。

最小函数依赖集

例：设F = {AB→C，A→B},求Fm。

解:

Fm = {B→C，A→B}

对不对？为什么?

不对！上面的F与Fm根本不等价。

Fm = {A→C，A→B}