UNIVERSIDAD DE GRANADA

/ UGR / decsai

Departamento de Ciencias de la
Computación e Inteligencia Artificial

Enunciado

Sea R una relación de esquema {A, B, C, D, E, F} que verifica el conjunto F de dependencias funcionales {AD→EB, F→A, B→A, AB→D}. Determinar todas las claves candidatas de la relación.

Resolución

En primer lugar, sería recomendable establecer los tipos de atributos en función de si aparecen a la izquierda en todas las dependencias funcionales en las que aparece, a la derecha en todas las dependencias funcionales en las que aparece, a la izquierda de alguna y a la derecha en otra o en ninguna dependencia funcional:

- *atributos independientes* (no aparecen en ninguna dependencia): {C}
- *atributos equivalentes*: ninguno
- *atributos que aparecen a la izquierda en todas las dependencias en las que aparecen*: {F}
- *atributos que aparecen a la izquierda de alguna dependencia funcional y a la derecha de otra*: {A, B y D}
- *atributos que aparecen a la derecha en todas las dependencias en las que aparecen*: {E}

Establecidos los tipos, podemos aplicar el algoritmo de extracción de claves candidatas:

1º Eliminación de atributos independientes:

R_{SI} = R – {C} = {A, B, D, E, F}

2º Eliminación de equivalencias:

No existen equivalencias entre atributos, por lo que R_{SIE}= R_{SI}= {A, B, D, E, F} y F_{SIE} = {AD→EB, F→A, B→A, AB→D}

3º Procesamiento de atributos que aparecen sólo a la izquierda:

K_p = {F} = F

Puesto que disponemos de un candidato, debemos comprobar si es clave, generando el cierre transitivo de atributos:

K_p^+= {F, A} ≠ R_{SIE} por lo que F no es una clave pero debe participar en todas las claves, por lo que es necesario proceder al paso 4.

4º Procesamiento de atributos que aparecen a la izquierda y a la derecha:

Para ello, construimos un conjunto de candidatos tomando el candidato del paso 3 y lo combinamos con todos los atributos que estén a la izquierda en alguna dependencia y a la derecha en otra, y no hayan sido generado por el candidato del paso 3. Es decir, habría que combinar el conjunto {F} (del paso 3) con los atributos B y D (F no se combina con A porque éste último aparece en el cierre transitivo de F, es decir, es generado por F, por lo que pueden combinarse).

K_p ’= {FB, FD}

Al haber candidatos, comprobamos si el primero es clave, calculando su cierre transitivo:

FB^+ = {F, B, A, D, E} = R_{SIE} por lo que FB es una clave para R_{SIE}, CK_{SIE} = {FB} y K_p ’= {FD}

Procedemos con el candidato FD:

FD^+ = {F, D, A, E, B} = R_{SIE} por lo que FD es una clave para R_{SIE}, CK_{SIE} = {FB, FD} y K_p ’= {}

Al no quedar candidatos que explorar, damos por concluido el paso 4.

5º Incorporación de atributos independientes:

Puesto que hay atributos independientes que fueron aislados en el paso 1, este paso modifica las claves obtenidas incorporando los atributos independientes a todas las claves, por lo que CK' = {FBC, FCD}.

6º Incorporación de atributos equivalentes:

Al no existir equivalencias entre atributos, este paso no modifica las claves obtenidas, por lo que CK = {FBC, FCD}

Enunciado

Sea R una relación de esquema {A, B, C, D, E}, con un conjunto CK de claves candidatas formada por una sola clave {E} y que verifica el siguiente conjunto de dependencias funcionales:

$$\{E \rightarrow C, C \rightarrow D, C \rightarrow A, ED \rightarrow B\}$$

a) obtener el recubrimiento minimal del conjunto de dependencias, y
b) utilizar dicho recubrimiento minimal para determinar si la relación está en Forma Normal de Boyce y Codd, asumiendo que está en segunda y tercera forma normal sin comprobarlo. Si no está en Forma Normal de Boyce y Codd, realizar una descomposición hasta que todas las relaciones que integran la descomposición estén en dicha forma normal. **Nota**: Si no calcula previamente el recubrimiento minimal, realice este apartado con el conjunto de dependencias inicial.

Resolución

Apartado (a)

El algoritmo de cálculo del recubrimiento minimal de dependencias funcionales F' nos permite obtener un conjunto de dependencias funcionales que es equivalente a F (se pueden deducir las mismas propiedades que las que se deducen de F aplicando los axiomas y las reglas de Armstrong un número finito de veces).

El algoritmo consta de tres pasos, que se detallan a continuación.

El **primer paso** construye el conjunto $\mathbf{F^{(1)}}$ en el que se simplifican las partes derechas de las dependencias de F aplicando la regla de descomposición a cada una de las dependencias que tengan varios atributos en su parte derecha. Dado que ninguna dependencia de F tiene una parte derecha compuesta, $F^{(1)}$=F.

El **segundo paso** construye el conjunto $\mathbf{F^{(2)}}$ en el que se simplifican las partes izquierdas de las dependencias de F buscando la existencia de dependencias entre atributos de la parte izquierda (a lo que se denominan *atributos extraños* o *raros*) de modo que se pueda aplicar la regla de pseudo-transitividad para reducir el número de atributos en la parte izquierda.

Análizamos la dependencia $ED \rightarrow B$ en busca de atributos raros, de modo que debemos examinar si E es raro con respecto a D y viceversa.

Se dice que E es raro con respecto a D si, y sólo si, $E \rightarrow D$ pertenece al conjunto F o se puede deducir de él (es decir, si pertenece a F^+). Como sabemos, la obtención de esta dependencia puede ser bastante complicada pero podemos saber si se puede deducir de F. La citada dependencia pertenece a F^+ si, y sólo si, se cumple que $D \in E^+$. Como podemos comprobar $E^+ = \{E, C, D, A, B\}$ y $D \in E^+$, por lo que D es raro con respecto a E y podemos aplicar la regla de pseudo-transitividad para eliminar D de este conjunto de atributos, quedando la dependencia como sigue $E \rightarrow B$ en $F^{(2)}$.

En resumen, nos queda un $F^{(2)}=\{E \rightarrow C, C \rightarrow D, C \rightarrow A, E \rightarrow B\}$.

El **tercer paso** construye el conjunto $\mathbf{F^{(3)}}$, en el que se simplifican dependencias completas, eliminándolas por ser redundantes (deducibles por combinación de otras). Para ello, procesamos cada dependencia eliminándola del conjunto considerado y tratando de deducirla de nuevo de las restantes. Como hemos visto, tratar de deducir una dependencia puede llegar a ser tedioso, pero basta con calcular el cierre de atributos de la parte izquierda de la dependencia con respecto a las dependencias restantes (menos la eliminada). Si la parte derecha de la dependencia está incluida en el cierre, la dependencia es deducible del conjunto de las restantes, incluso después de haberla eliminado, por lo que es redundante y puede eliminarse de $F^{(3)}$ definitivamente.

Empezamos con la evaluación de la posible redundancia de $E \rightarrow C$. Evaluamos el cierre E^+ con respecto al conjunto de dependencias restante $\{C \rightarrow D, C \rightarrow A, E \rightarrow B\}$ y obtenemos $E^+=\{E, B\}$. Dado que $C \notin E^+$, la dependencia $E \rightarrow C$ no puede deducirse de las demás y no es redundante.

Procedemos con la evaluación de la posible redundancia de $C \rightarrow D$. Después de calcular el cierre C^+ con respecto al conjunto de depentencias restante $\{E \rightarrow C, C \rightarrow A, E \rightarrow B\}$, observamos que $D \notin C^+ = \{C, A\}$ por lo que la dependencia $C \rightarrow D$ no puede deducirse de las demás y no es redundante.

Para ver si $C \rightarrow A$ es redundante, basta con calcular C^+ con respecto al conjunto de dependencias restante $\{E \rightarrow C, C \rightarrow D, E \rightarrow B\}$. Como podemos observar, $A \notin C^+ = \{C, D\}$ por lo que la dependencia $C \rightarrow A$ no puede deducirse de las demás y no es redundante.

Por último, para ver si $E \rightarrow B$ es redundante, basta con calcular E^+ con respecto al conjunto de dependencias restante $\{E \rightarrow C, C \rightarrow D, C \rightarrow A\}$. Como podemos observar, $B \notin E^+ = \{E, C\}$ por lo que la dependencia $E \rightarrow B$ no puede deducirse de las demás y no es redundante.

Después de evaluar cada dependencia del conjunto $F^{(2)}$, y decidir que ninguna es redundante, nos queda un conjunto $F^{(3)} = F^{(2)} = \{E \rightarrow C, C \rightarrow D, C \rightarrow A, E \rightarrow B\}$.

De modo que el recubrimiento minimal de F es $F' = F^{(3)} = \{E \rightarrow C, C \rightarrow D, C \rightarrow A, E \rightarrow B\}$

Apartado (b)

Normalización con respecto a F’

La relación R esté en Forma Normal de Boyce y Codd si, y sólo si, todas las dependencias tienen a su izquierda una clave candidata. Según esta norma, podemos decir que la relación no está en BCNF porque $C \rightarrow D$ y $C \rightarrow A$ forman parte de F’ y C no es clave candidata de la relación.

Para conseguir una normalización, debemos aplicar el Teorema de Heath sobre una de las dependencias que no cumplen la forma normal.

Empezamos normalizando por $C \rightarrow D$:

$R_1 = \{C, D\}$, $F_1 = \{C \rightarrow D\}$, $CK_1 = \{C\}$, **R_1 está en FNBC** puesto que todas las dependencias funcionales de F_1 tienen una clave candidata de R_1 a su izquierda.

$R_2 = \{A, B, C, E\}$, $F_2 = \{E \rightarrow C, C \rightarrow A, E \rightarrow B\}$, $CK_2 = \{E\}$, R_2 no está en BCNF porque $C \rightarrow A$ forma parte de F_2 pero C no es clave candidata de R_2.

Es necesario normalizar la relación R_2 mediante la aplicación del Teorema de Heath aplicado sobre $C \rightarrow A$:

$R_{2,1} = \{C, A\}$, $F_2 = \{C \rightarrow A\}$, $CK_2 = \{C\}$, **$R_{2,1}$ está en BCNF** porque todas sus dependencias tienen a la izquierda una llave candidata de $R_{2,1}$.

$R_{2,2} = \{B, C, E\}$, $F_2 = \{E \rightarrow C, E \rightarrow B\}$, $CK_2 = \{E\}$, **$R_{2,2}$ está en BCNF** porque todas sus dependencias tienen a la izquierda una llave candidata de $R_{2,2}$.

El resultado de nuestra normalización es el conjunto de relaciones:

$$\{(\{C, D\}, r_1), (\{C, A\}, r_{2,1}), (\{B, C, E\}, r_{2,2})\}$$

Normalización con respecto a F (alternativa 1)

La relación R esté en Forma Normal de Boyce y Codd si, y sólo si, todas las dependencias tienen a su izquierda una clave candidata. Según esta norma, podemos decir que la relación no está en BCNF porque $C \rightarrow D$, $C \rightarrow A$ y $ED \rightarrow B$ forman parte de F y ni C ni ED son claves candidatas de la relación.

Para conseguir una normalización, debemos aplicar el Teorema de Heath sobre una de las dependencias que no cumplen la forma normal.

Empezamos normalizando por $C \rightarrow D$:

$R_1 = \{C, D\}$, $F_1 = \{C \rightarrow D\}$, $CK_1 = \{C\}$, **R_1 está en FNBC** puesto que todas las dependencias funcionales de F_1 tienen una clave candidata de R_1 a su izquierda.

$R_2 = \{A, B, C, E\}$, $F_2 = \{E \rightarrow C, C \rightarrow A\}$, $CK_2 = \{E\}$, R_2 no está en BCNF porque $C \rightarrow A$ forma parte de F_2 pero C no es clave candidata de R_2.

Es necesario normalizar la relación R_2 mediante la aplicación del Teorema de Heath aplicado sobre $C \rightarrow A$:

$R_{2,1} = \{C, A\}$, $F_2 = \{C \rightarrow A\}$, $CK_2 = \{C\}$, **$R_{2,1}$ está en BCNF** porque todas sus dependencias tienen a la izquierda una llave candidata de $R_{2,1}$.

$R_{2,2} = \{B, C, E\}$, $F_2 = \{E \rightarrow C\}$, $CK_2 = \{E\}$, **$R_{2,2}$ está en BCNF** porque todas sus dependencias tienen a la izquierda una llave candidata de $R_{2,2}$.

El resultado de nuestra normalización es el conjunto de relaciones:

$$\{(\{C, D\}, r_1), (\{C, A\}, r_{2,1}), (\{B, C, E\}, r_{2,2})\}$$

Normalización con respecto a F (alternativa 2)

La relación R esté en Forma Normal de Boyce y Codd si, y sólo si, todas las dependencias tienen a su izquierda una clave candidata. Según esta norma, podemos decir que la relación no está en BCNF porque $C \rightarrow D$, $C \rightarrow A$ y $ED \rightarrow B$ forman parte de F y ni C ni ED son claves candidatas de la relación.

Para conseguir una normalización, debemos aplicar el Teorema de Heath sobre una de las dependencias que no cumplen la forma normal.

Empezamos normalizando por $ED \rightarrow B$:

$R_1 = \{B, D, E\}$, $F_1 = \{ED \rightarrow B\}$, $CK_1 = \{ED\}$, **R_1 está en FNBC** puesto que todas las dependencias funcionales de F_1 tienen una clave candidata de R_1 a su izquierda.

$R_2 = \{A, C, D, E\}$, $F_2 = \{E \rightarrow C, C \rightarrow D, C \rightarrow A\}$, $CK_2 = \{E\}$, R_2 no está en BCNF porque $C \rightarrow D$ y $C \rightarrow A$ forman parte de F_2 pero C no es clave candidata de R_2.

Es necesario normalizar la relación R_2 mediante la aplicación del Teorema de Heath aplicado sobre $C \rightarrow D$:

$R_{2,1} = \{C, D\}$, $F_2 = \{C \rightarrow D\}$, $CK_2 = \{C\}$, **$R_{2,1}$ está en BCNF** porque todas sus dependencias tienen a la izquierda una llave candidata de $R_{2,1}$.

$R_{2,2} = \{A, C, E\}$, $F_2 = \{E \rightarrow C, C \rightarrow A\}$, $CK_2 = \{E\}$, $R_{2,2}$ no está en BCNF porque $C \rightarrow A$ pertenece a F_2 y C no es una llave candidata de $R_{2,2}$.

Es necesario normalizar la relación $R_{2,2}$ mediante la aplicación del Teorema de Heath aplicado sobre $C \rightarrow A$:

$R_{2,2,1} = \{A, C\}$, $F_2 = \{C \rightarrow A\}$, $CK_2 = \{C\}$, **$R_{2,2,1}$ está en BCNF** porque todas sus dependencias tienen a la izquierda una llave candidata de $R_{2,2,1}$.

$R_{2,2} = \{C, E\}$, $F_2 = \{E \rightarrow C\}$, $CK_2 = \{E\}$, **$R_{2,2,2}$ está en BCNF** porque todas sus dependencias tienen a la izquierda una llave candidata de $R_{2,2,2}$.

El resultado de nuestra normalización es el conjunto de relaciones:

$$\{(\{B, D, E\}, r_1), (\{C, D\}, r_{2,1}), (\{A, C\}, r_{2,2,1}), (\{C, E\}, r_{2,2,2})\}$$