Database Management Systems

Module 24: Relational Database Design/4

Partha Pratim Das

Department of Computer Science and Engineering
Indian Institute of Technology, Kharagpur

ppd@cse.iitkgp.ac.in

Module Recap

- Introduced the theory of functional dependencies
- Discussed issues in "good" design in the context of functional dependencies

Module Objectives

- To Learn Algorithms for Properties of Functional Dependencies

Module Outline

- Algorithms for Functional Dependencies

Algorithms for Functional Dependencies

Attribute Set Closure

- $R = (A, B, C, G, H, I)$
- $F = \{A \rightarrow B,\ A \rightarrow C,\ CG \rightarrow H,\ CG \rightarrow I,\ B \rightarrow H\}$
- $(AG)^+$
  a) $result = AG$
  b) $result = ABCG$ $\quad (A \rightarrow C$ and $A \rightarrow B)$
  c) $result = ABCGH$ $\quad (CG \rightarrow H$ and $CG \subseteq AGBC)$
  d) $result = ABCGHI$ $\quad (CG \rightarrow I$ and $CG \subseteq AGBCH)$
- Is AG a candidate key?
  a) Is AG a super key?
    i) Does $AG \rightarrow R? ==$ Is $(AG)^+ \supseteq R$
  b) Is any subset of AG a superkey?
    i) Does $A \rightarrow R? ==$ Is $(A)^+ \supseteq R$
    ii) Does $G \rightarrow R? ==$ Is $(G)^+ \supseteq R$

Attribute Set Closure: Uses

There are several uses of the attribute closure algorithm:

- Testing for superkey:
  - To test if α is a superkey, we compute α^+, and check if α^+ contains all attributes of R.
- Testing functional dependencies
  - To check if a functional dependency $\alpha \to \beta$ holds (or, in other words, is in F^+), just check if $\beta \subseteq \alpha^+$.
  - That is, we compute α^+ by using attribute closure, and then check if it contains β.
  - Is a simple and cheap test, and very useful
- Computing closure of F
  - For each $\gamma \subseteq R$, we find the closure γ^+, and for each $S \subseteq \gamma^+$, we output a functional dependency $\gamma \to S$.

Extraneous Attributes

- Consider a set F of FDs and the FD $\alpha \rightarrow \beta$ in F.
  - Attribute A is **extraneous** in α if $A \in \alpha$ and F logically implies $(F - \{\alpha \rightarrow \beta\}) \cup \{(\alpha - A) \rightarrow \beta\}$.
  - Attribute A is **extraneous** in β if $A \in \beta$ and the set of FDs $(F - \{\alpha \rightarrow \beta\}) \cup \{\alpha \rightarrow (\beta - A)\}$ logically implies F.
- *Note:* Implication in the opposite direction is trivial in each of the cases above, since a "stronger" functional dependency always implies a weaker one
- Example: Given $F = \{A \rightarrow C, AB \rightarrow C\}$
  - B is extraneous in $AB \rightarrow C$ because $\{A \rightarrow C, AB \rightarrow C\}$ logically implies $A \rightarrow C$ (that is, the result of dropping B from $AB \rightarrow C$).
  - $A^+ = AC$ in $\{A \rightarrow C, AB \rightarrow C\}$
- Example: Given $F = \{A \rightarrow C, AB \rightarrow CD\}$
  - C is extraneous in $AB \rightarrow CD$ since $AB \rightarrow C$ can be inferred even after deleting C
  - $AB^+ = ABCD$ in $\{A \rightarrow C, AB \rightarrow D\}$

Extraneous Attributes (2): Tests

- Consider a set F of functional dependencies and the functional dependency $\alpha \rightarrow \beta$ in F.
- To test if attribute $A \in \alpha$ is extraneous in α
  a) Compute $(\{\alpha\} - A)^+$ using the dependencies in F
  b) Check that $(\{\alpha\} - A)^+$ contains β; if it does, A is extraneous in α
- To test if attribute $A \in \beta$ is extraneous in β
  a) Compute α^+ using only the dependencies in
     $F' = (F - \{\alpha \rightarrow \beta\}) \cup \{\alpha \rightarrow (\beta - A)\}$,
  b) Check that α^+ contains A; if it does, A is extraneous in β

Equivalence of Sets of Functional Dependencies

- Let F & G are two functional dependency sets.
  - These two sets F & G are equivalent if $F^+ = G^+$. That is: $(F^+ = G^+) \Leftrightarrow (F^+ \Rightarrow G \text{ and } G^+ \Rightarrow F)$
  - Equivalence means that every functional dependency in F can be inferred from G, and every functional dependency in G an be inferred from F
- F and G are equal only if
  - F covers G: Means that all functional dependency of G are logically members of functional dependency set $F \Rightarrow F^+ \supseteq G$.
  - G covers F: Means that all functional dependency of F are logically members of functional dependency set $G \Rightarrow G^+ \supseteq F$.

Condition	**CASES**			
F Covers G	True	True	False	False
G Covers F	True	False	True	False
Result	F=G	F⊃G	G⊃F	No Comparison

Canonical Cover

- Sets of FDs may have redundant dependencies that can be inferred from the others
- Can we have some kind of "optimal" or "minimal" set of FDs wto work with?
- A **Canonical Cover** for F is a set of dependencies F_c such that ALL the following properties are satisfied:
  - $F^+ = F_c^+$. Or,
    - ▷ F logically implies all dependencies in F_c
    - ▷ F_c logically implies all dependencies in F
  - No functional dependency in F_c contains an extraneous attribute
  - Each left side of functional dependency in F_c is unique. That is, there are no two dependencies $\alpha_1 \rightarrow \beta_1$ and $\alpha_2 \rightarrow \beta_2$ in such that $\alpha_1 \rightarrow \alpha_2$
- Intuitively, a **Canonical cover** of F is a **minimal** set of FDs
  - Equivalent to F
  - Having no redundant FDs
  - No redundant parts of FDs
- **Minimal / Irreducible Set of Functional Dependencies**

Canonical Cover (2): Example

- For example: $A \to C$ is redundant in: $\{A \to B, B \to C, A \to C\}$
- Parts of a functional dependency may be redundant
  - For example: on RHS: $\boldsymbol{\{A \to B, B \to C, A \to CD\}}$ can be simplified to $\boldsymbol{\{A \to B, B \to C, A \to D\}}$
    - In the forward: (1) $A \to CD \Rightarrow A \to C$ and $A \to D$
      (2) $A \to B, B \to C \Rightarrow A \to C$
    - In the reverse: (1) $A \to B, B \to C \Rightarrow A \to C$
      (2) $A \to C, A \to D \Rightarrow A \to CD$
  - For example: on LHS: $\boldsymbol{\{A \to B, B \to C, AC \to D\}}$ can be simplified to $\boldsymbol{\{A \to B, B \to C, A \to D\}}$
    - In the forward: (1) $A \to B, B \to C \Rightarrow A \to C \Rightarrow A \to AC$
      (2) $A \to AC, AC \to D \Rightarrow A \to D$
    - In the reverse: $A \to D \Rightarrow AC \to D$

Canonical Cover (3): RHS

- $\{A \rightarrow B, B \rightarrow C, A \rightarrow CD\} \Rightarrow \{A \rightarrow B, B \rightarrow C, A \rightarrow D\}$
  - (1) $A \rightarrow CD \Rightarrow A \rightarrow C$ and $A \rightarrow D$
    (2) $A \rightarrow B, B \rightarrow C \Rightarrow A \rightarrow C$
  - $A^+ = ABCD$
- $\{A \rightarrow B, B \rightarrow C, A \rightarrow D\} \Rightarrow \{A \rightarrow B, B \rightarrow C, A \rightarrow CD\}$
  - $A \rightarrow B, B \rightarrow C \Rightarrow A \rightarrow C$
  - $A \rightarrow C, A \rightarrow D \Rightarrow A \rightarrow CD$
  - $A^+ = ABCD$

Canonical Cover (4): LHS

- $\{A \rightarrow B, B \rightarrow C, AC \rightarrow D\} \Rightarrow \{A \rightarrow B, B \rightarrow C, A \rightarrow D\}$
  - $A \rightarrow B, B \rightarrow C \Rightarrow A \rightarrow C \Rightarrow A \rightarrow AC$
  - $A \rightarrow AC, AC \rightarrow D \Rightarrow A \rightarrow D$
  - $A^+ = ABCD$
- $\{A \rightarrow B, B \rightarrow C, A \rightarrow D\} \Rightarrow \{A \rightarrow B, B \rightarrow C, AC \rightarrow D\}$
  - $A \rightarrow D \Rightarrow AC \rightarrow D$
  - $AC^+ = ABCD$

Canonical Cover (5)

- To compute a canonical cover for F:

  repeat

  Use the union rule to replace any dependencies in F

  $\alpha_1 \rightarrow \beta_1$ and $\alpha_1 \rightarrow \beta_2$ with $\alpha_1 \rightarrow \beta_1\beta_2$

  Find a functional dependency $\alpha \rightarrow \beta$ with an extraneous attribute either in α or in β

  /* Note: test for extraneous attributes done using F_c, not F */

  If an extraneous attribute is found, delete it from $\alpha \rightarrow \beta$

  until F does not change

- Note: Union rule may become applicable after some extraneous attributes have been deleted, so it has to be re-applied

Canonical Cover (6): Example

- $R = (A, B, C)$
  $F = \{A \rightarrow BC, B \rightarrow C, A \rightarrow B, AB \rightarrow C\}$
- Combine $A \rightarrow BC$ and $A \rightarrow B$ into $A \rightarrow BC$
  - Set is now $\{A \rightarrow BC, B \rightarrow C, AB \rightarrow C\}$
- A is extraneous in $AB \rightarrow C$
  - Check if the result of deleting A from $AB \rightarrow C$ is implied by the other dependencies
    - ▷ Yes: in fact, $B \rightarrow C$ is already present!
  - Set is now $\{A \rightarrow BC, B \rightarrow C\}$
- C is extraneous in $A \rightarrow BC$
  - Check if $A \rightarrow C$ is logically implied by $A \rightarrow B$ and the other dependencies
    - ▷ Yes: using transitivity on $A \rightarrow B$ and $B \rightarrow C$.
      - Can use attribute closure of A in more complex cases
- The canonical cover is: $A \rightarrow B$, $B \rightarrow C$

Practice Problems on Functional Dependencies

- **Find if a given functional dependency is implied from a set of Functional Dependencies:**
  a) For: $A \rightarrow BC, CD \rightarrow E, E \rightarrow C, D \rightarrow AEH, ABH \rightarrow BD, DH \rightarrow BC$
     i) Check: $BCD \rightarrow H$
     ii) Check: $AED \rightarrow C$
  b) For: $AB \rightarrow CD, AF \rightarrow D, DE \rightarrow F, C \rightarrow G, F \rightarrow E, G \rightarrow A$
     i) Check: $CF \rightarrow DF$
     ii) Check: $BG \rightarrow E$
     iii) Check: $AF \rightarrow G$
     iv) Check: $AB \rightarrow EF$
  c) For: $A \rightarrow BC, B \rightarrow E, CD \rightarrow EF$
     i) Check: $AD \rightarrow F$

Practice Problems on Functional Dependencies (2)

- **Find Super Key using Functional Dependencies:**
  a) Relational Schema $R(ABCDE)$. Functional dependencies: $AB \rightarrow C, DE \rightarrow B, CD \rightarrow E$
  b) Relational Schema $R(ABCDE)$. Functional dependencies: $AB \rightarrow C, C \rightarrow D, B \rightarrow EA$

Practice Problems on Functional Dependencies (3)

- **Find Candidate Key using Functional Dependencies:**
  a) Relational Schema $R(ABCDE)$. Functional dependencies: $AB \rightarrow C, DE \rightarrow B, CD \rightarrow E$
  b) Relational Schema $R(ABCDE)$. Functional dependencies: $AB \rightarrow C, C \rightarrow D, B \rightarrow EA$

- **Find Prime and Non Prime Attributes using Functional Dependencies:**
  a) $R(ABCDEF)$ having FDs $\{AB \rightarrow C, C \rightarrow D, D \rightarrow E, F \rightarrow B, E \rightarrow F\}$
  b) $R(ABCDEF)$ having FDs $\{AB \rightarrow C, C \rightarrow DE, E \rightarrow F, C \rightarrow B\}$
  c) $R(ABCDEFGHIJ)$ having FDs $\{AB \rightarrow C, A \rightarrow DE, B \rightarrow F, F \rightarrow GH, D \rightarrow IJ\}$
  d) $R(ABDLPT)$ having FDs $\{B \rightarrow PT, A \rightarrow D, T \rightarrow L\}$
  e) $R(ABCDEFGH)$ having FDs $\{E \rightarrow G, AB \rightarrow C, AC \rightarrow B, AD \rightarrow E, B \rightarrow D, BC \rightarrow A\}$
  f) $R(ABCDE)$ having FDs $\{A \rightarrow BC, CD \rightarrow E, B \rightarrow D, E \rightarrow A\}$
  g) $R(ABCDEH)$ having FDs $\{A \rightarrow B, BC \rightarrow D, E \rightarrow C, D \rightarrow A\}$

- **Prime Attributes**: Attribute set that belongs to any candidate key are called Prime Attributes
  - It is union of all the candidate key attribute: $\{CK_1 \cup CK_2 \cup CK_3 \cup \cdots\}$
  - If Prime attribute determined by other attribute set, then more than one candidate key is possible.
  - For example, If A is Candidate Key, and $X \rightarrow A$, then, X is also Candidate Key.
- **Non Prime Attribute**: Attribute set does not belong to any candidate key are called Non Prime Attributes

Practice Problems on Functional Dependencies (5)

- **Check the Equivalence of a Pair of Sets of Functional Dependencies:**
  a) Consider the two sets F and G with their FDs as below :
     i) $F : A \rightarrow C, AC \rightarrow D, E \rightarrow AD, E \rightarrow H$
     ii) $G : A \rightarrow CD, E \rightarrow AH$
  b) Consider the two sets P and Q with their FDs as below :
     i) $P : A \rightarrow B, AB \rightarrow C, D \rightarrow ACE$
     ii) $Q : A \rightarrow BC, D \rightarrow AE$

Practice Problems on Functional Dependencies (6)

- **Find the Minimal Cover or Irreducible Sets or Canonical Cover of a Set of Functional Dependencies:**
  a) $AB \rightarrow CD, BC \rightarrow D$
  b) $ABCD \rightarrow E, E \rightarrow D, AC \rightarrow D, A \rightarrow B$

Module Summary

- Studied Algorithms for Properties of Functional Dependencies

Slides used in this presentation are borrowed from http://db-book.com/ with kind permission of the authors.
Edited and new slides are marked with “PPD”.