CSE414 Database HW2

Fatih Selim YAKAR
161044054

1-) $R = \{A1, A2, A3, A4, A5, A6, A7, A8\}$

$F = \{ A_1A_3 \to A_7,$

$A_4 \to A_5A_7,$ → we can write that like $A_4 \to A_5$, $A_4 \to A_7$

$A_2A_3 \to A_4,$

$A_3A_7 \to A_2A_4,$ → we can write that like $A_3A_7 \to A_2$, $A_3A_7 \to A_4$

$A_1A_3A_4 \to A_2,$

$A_3A_5 \to A_1A_7 \}.$ → we can write that like $A_3A_5 \to A_1$, $A_3A_5 \to A_7$

- Extraneous Attributes

→ checking A_1 for $A_1A_3 \to A_7$

then we will find the A_3^+

$A_3^+ = A_3 \not\supseteq A_7$ so A_1 is necessary

→ checking A_3 for $A_1A_3 \to A_7$

then we will find the A_1^+

$A_1^+ = A_1 \not\supseteq A_7$ so A_3 is necessary

→ $A_4 \to A_5$ and $A_4 \to A_7$ are necessary because there is one attribute right and left side.

→ checking A_2 for $A_2A_3 \to A_4$

then we will find the A_3^+

$A_3^+ = A_3 \not\supseteq A_4$ so A_2 is necessary

→ checking A_3 for $A_2A_3 \to A_4$

then we will find the A_2^+

$A_2^+ = A_2 \not\supseteq A_4$ so A_3 is necessary

→ checking A_3 for $A_3A_7 \to A_2$

then we will find the. A_7^+

$A_7^+ = A_7 \not\supseteq A_2$ so A_3 is necessary

→ checking A_7 for $A_3A_7 \to A_2$

then we will find the A_3^+

$A_3^+ = A_3 \not\supseteq A_2$ so A_7 is necessary

→ checking A_3, A_7 for $A_3A_7 \to A_4$ like above.

$A_3^+ = A_3 \not\supseteq$
$A_7^+ = A_7 \not\supseteq$ A_4 so these are necessary.

→ checking A_4 for $A_1A_3A_4 \to A_2 \Rightarrow A_1A_3 \to A_2$

$A_1A_3^+ \to A_1A_3$

$(A_1A_3 \to A_7)\ A_1A_3A_7$

$(A_3A_7 \to A_2)\ A_1A_3A_7A_2 \supseteq A_2$ ✓ so A_4 is extraneous//

⋮ ⋮

→ Checking A_3 for $A_3A_5 \to A_1$

then we will find the A_5^+

$A_5^+ = A_5 \not\supseteq A_1$ so A_3 is necessary

→ Checking A_5 for $A_3A_5 \to A_1$

then we will find the A_3^+

$A_3^+ = A_3 \not\supseteq A_1$ so A_5 is necessary

Summary for Extraneous Attributes

The solutions here were made by the following method indicated on the slide.

- Let R be relation schema and let F be a set of functional dependencies that hold on R. Consider an attribute in the functional dependency $\alpha \to \beta$
- To test if attribute $A \in \beta$ is extraneous in β
  - Consider the set $F' = (F - \{\alpha \to \beta\}) \cup \{\alpha \to (\beta - A)\}$
  - check that α^+ contains A; if it does A is extraneous in β
- To test if attribute $A \in \alpha$ is extraneous in α
  - Let $\gamma = \alpha - \{A\}$. Check if $\gamma \to \beta$ can be inferred from F
    - Compute γ^+
    - If γ^+ includes all attributes in β then A is extraneous in α

- **Computing Canonical Cover**

A canonical cover for F is a set of dependencies F_c such that

- F logically implies all dependencies in F_c, and
- F_c logically implies all dependencies in F, and
- No functional dependency in F_c contains an extraneous attribute, and
- Each left side of functional dependency in F_c is unique. That is, there are no two dependencies in F_c.
  - $\alpha_1 \to \beta_1$ and $\alpha_2 \to \beta_2$ such that
  - $\alpha_1 = \alpha_2$

So we find the left hand side's closure (except itself) then if we found the right hand side value in the closure, the FD is redundant.

→ FDs: $\{A_1A_3 \to A_7,$ -
$A_4 \to A_5,$
$A_4 \to A_7,$
$A_2A_3 \to A_4,$
$A_3A_7 \to A_2,$ -
$A_3A_7 \to A_4,$
$A_1A_3 \to A_2,$
$A_3A_5 \to A_1,$
$A_3A_5 \to A_7\}$ -

→ $A_1A_3 \to A_7$

$A_1A_3^+ = A_1A_3$
$A_1A_3A_2$ $(A_1A_3 \to A_2)$
$A_1A_3A_2A_4$ $(A_2A_3 \to A_4)$
$A_1A_3A_2A_4A_5$ $(A_4 \to A_5)$
$A_1A_3A_2A_4A_5A_7$ $(A_3A_5 \to A_7)$

covers A_7 ⇒ so $A_1A_3 \to A_7$ is redundant

→ $A_4 \to A_5$ and $A_4 \to A_7$ are necessary.

$A_4^+ = A_4A_7$ (except itself)
$A_4A_7 \not\supseteq A_5$
$A_4^+ = A_4A_5$ (except itself)
$A_4A_5 \supseteq A_7$

→ $A_2A_3 \to A_4$

$A_2A_3^+ = A_2A_3 \not\supseteq A_4$
the FD is necessary

→ $A_3A_7 \to A_2$

$A_3A_7^+ = A_3A_7$
$A_3A_7A_4$ $(A_3A_7 \to A_4)$
$A_3A_7A_4A_5$ $(A_4 \to A_5)$
$A_3A_7A_4A_5A_1$ $(A_3A_5 \to A_1)$
$A_3A_7A_4A_5A_1A_2$ covers A_2 $(A_1A_3 \to A_2)$
so $A_3A_7 \to A_2$ is redundant

→ $A_3A_7 \to A_4$

$A_3A_7^+ = A_3A_7 \not\supseteq A_4$
the FD is necessary

→ $A_1A_3 \to A_2$

$A_1A_3^+ = A_1A_3 \not\supseteq A_2$
the FD is necessary

→ $A_3A_5 \to A_1$

$A_3A_5^+ = A_3A_5$
$A_3A_5A_7$ $(A_3A_5 \to A_7)$
$A_3A_5A_7A_4 \not\supseteq A_1$ $(A_3A_7 \to A_4)$
the FD is necessary

→ $A_3A_5 \to A_7$

$A_3A_5^+ = A_3A_5$
$A_3A_5A_1$ $(A_3A_5 \to A_1)$
$A_3A_5A_1A_2$ $(A_1A_3 \to A_2)$
$A_3A_5A_1A_2A_4$ $(A_2A_3 \to A_4)$
$A_3A_5A_1A_2A_4A_7 \supseteq A_7$ $(A_4 \to A_7)$
so $A_3A_5 \to A_7$ is redundant

Canonical Cover → $F_c = \{A_1A_3 \to A_2,\ A_4 \to A_5A_7,$
$A_2A_3 \to A_4,\ A_3A_7 \to A_4,$
$A_3A_5 \to A_1\}$ //

2-) a- $R(A_1, A_2, A_3, A_4, A_5, A_6)$

$F = \{ A_1, A_2 \rightarrow A_3,$
$A_1, A_4 \rightarrow A_5,$
$A_2 \rightarrow A_4,$
$A_1, A_6 \rightarrow A_2 \}$

$\rightarrow A_1A_2^+ = A_1A_2$
$A_1A_2A_3 \; (A_1A_2 \rightarrow A_3)$
$A_1A_2A_3A_4 \; (A_2 \rightarrow A_4)$
$\underline{A_1A_2A_3A_4A_5} \; (A_1A_4 \rightarrow A_5)$

$\rightarrow A_1A_6^+ = A_1A_6$
$A_1A_6A_2 \; (A_1A_6 \rightarrow A_2)$
$A_1A_6A_2A_3 \; (A_1A_2 \rightarrow A_3)$
$A_1A_6A_2A_3A_4 \; (A_2 \rightarrow A_4)$
$\underline{A_1A_6A_2A_3A_4A_5} \; (A_1A_4 \rightarrow A_5)$

b- There are several uses of the attribute closure algorithm:

- Testing for superkeys and candidate keys
  - 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 $\alpha \rightarrow \beta$ holds, 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 $Y \subseteq R$, we find the closure Y^+, and for each $S \subseteq Y^+$, we output a functional dependency $Y \rightarrow S$.

3-) a- $R(A_1, A_2, A_3, A_4)$

$F = \{ A_1 \rightarrow A_2,$ $\Rightarrow$ A_1 and A_2 superkeys
$A_2 \rightarrow A_3 \}$

BCNF definition:

- A relation schema R is in BCNF with respect to a set F of functional dependencies if for all fd in F^+ of the form

$\alpha \rightarrow \beta$

where $\alpha \subseteq R$ and $\beta \subseteq R$, at least one of the following holds:
- $\alpha \rightarrow \beta$ is trivial ($\beta \subseteq \alpha$)
- α is a superkey for R

trivial function dependencies

$A_1A_2A_3A_4 \to A_1$
$A_1A_2A_3A_4 \to A_2$
$A_1A_2A_3A_4 \to A_3$
$A_1A_2A_3A_4 \to A_4$

$A_2A_3A_4 \to A_2$
$A_2A_3A_4 \to A_3$
$A_2A_3A_4 \to A_4$

$A_1A_2A_4 \to A_1$
$A_1A_2A_4 \to A_2$
$A_1A_2A_4 \to A_4$

$A_1A_2A_3 \to A_1$
$A_1A_2A_3 \to A_2$
$A_1A_2A_3 \to A_3$

$A_3A_4 \to A_3$
$A_3A_4 \to A_4$

$A_2A_4 \to A_2$
$A_2A_4 \to A_4$

$A_2A_3 \to A_2$
$A_2A_3 \to A_3$

$A_1A_4 \to A_1$
$A_1A_4 \to A_4$

$A_1A_3 \to A_1$
$A_1A_3 \to A_3$

$A_1A_2 \to A_1$
$A_1A_2 \to A_2$

$A_1 \to A_1$
$A_2 \to A_2$
$A_3 \to A_3$
$A_4 \to A_4$

Applies BCNF's trivial rule

non-trivial functional dependencies

$A_1 \to A_2$
$A_1 \to A_3$
$A_2 \to A_3$
$A_1A_2 \to A_3$
$A_1A_3 \to A_2$
$A_1A_4 \to A_2$
$A_1A_4 \to A_3$
$A_2A_4 \to A_3$

$A_1A_2A_4 \to A_3$

Applies BCNF's superkey rule with left hand sides because of the A_4A_2 superkey

b- $R1(A_1, A_2)$, $R_2(A_1, A_3)$, $R_3(A_1, A_4)$

$F^+ \Downarrow$

trivial: $A_1A_2 \to A_1$, $A_1A_2 \to A_2$
superkey: $A_1 \to A_2$
trivial: $A_1 \to A_1$, $A_2 \to A_2$

$F^+ \Downarrow$

tri: $A_1A_3 \to A_1$, $A_1A_3 \to A_3$
superkey: $A_1 \to A_3$
tri: $A_1 \to A_1$, $A_3 \to A_3$

$F^+ \Downarrow$

$A_1A_4 \to A_1$, $A_1A_4 \to A_4$: trivial
$A_1 \to A_1$, $A_4 \to A_4$: trivial

R_1 and R_2 and R_3 are in BCNF

c- A functional dependency $A_1 \to A_2$ is preserved in a relation R if R contains all the attributes of A_1 and A_2

$A_1 \to A_2$ is preserved in R_1
$A_2 \to A_3$ is not preserved in R_1

$A_1 \to A_2$ is not preserved in R_2
$A_2 \to A_3$ is not preserved in R_2

$A_1 \to A_2$ is not preserved in R_3
$A_2 \to A_3$ is not preserved in R_3

$R_1 \cap R_2 \cap R_3 = \{A_1\}$ $\qquad$ $F(A_1 \rightarrow A_2 ,\ A_2 \rightarrow A_3)$

$R_1 \Rightarrow A_1 \rightarrow A_1 A_2 \longrightarrow$

$R_2 \Rightarrow A_1 \rightarrow A_1 A_3 \longrightarrow$

$R_3 \Rightarrow A_1 \rightarrow A_1 A_4 \longrightarrow$

$A_1 \rightarrow A_2$ is preserved in R_1

$A_2 \rightarrow A_3$ is not preserved any relation so we have to calculate $R_1 \times R_2 \times R_3$ for find $A_2 \rightarrow A_3$. So there isn't dependency preserving here //