Tutorial 5: Logical Database Design

Jin, Ziyang
34893140
f4a0b

Kim, Joon Hyung
35183128
l1m8

February 2018

`StudentInfo(S, N, M, A, C, T, I, L, G)`

with the following FDs:

1. `S -> N`
2. `C -> T, I`
3. `I -> L`
4. `S, C, M -> G`
5. `S, M -> A`
6. `A -> M`

1 Find all keys and prove that you have found them all.

Since `S` and `C` do not appear in any RHS of the FDs,
`S` and `C` must be part of every key.
$(\texttt{S C})^+ = \texttt{S C N T I L} \neq \texttt{SI}$, so `(S C)` is not a key.

Guess `(S C M)` and `(S C A)` are keys.
$(\texttt{S C M})^+ = (\texttt{S C M N T I L G A}) = \texttt{SI}$ so `(S C M)` is a superkey.
$(\texttt{S C A})^+ = (\texttt{S C A N T I L G M}) = \texttt{SI}$ so `(S C A)` is a superkey.
Since `(S C)` is not a key, so `(S C M)` and `(S C A)` are minimal, so they are keys.

Consider the maximal set $\texttt{X} \subseteq (\texttt{S N M A C T I L G})$ such that $(\texttt{S C}) \subset \texttt{X}$ and $\texttt{A, M} \notin \texttt{X}$. The maximal such `X` is `(S N C T I L G)`
$\texttt{X}^+ = (\texttt{S N C T I L G})^+ = \texttt{S N C T I L G} \neq \texttt{SI}$
So any set other than `(S C M)` and `(S C A)` cannot be a key.

Therefore, `(S C M)` and `(S C A)` are the only keys for the relation `SI`.

2 Find a minimal cover for this set of FDs.

1. Put the FDs in a Standard Form.
   `S->N, C->T, C->I, I->L, SCM->G, SM->A, A->M.`
2. Minimize the Left Side of Each FD.
   Single attribute left sides are already minimized.
   `SCM->G`: `S,C` do not appear on LHS. `M` cannot be reduced from `S,C`.
   `SM->A`: `S` does not appear on LHS. `M` cannot be reduced from `S`.
   So the left sides are all minimized.
3. Delete Redundant FDs.
   `N, T, I, L, G, A, M` only appear once on RHS, so there is no redundant FD.

So the minimum cover for this set of FDs is:

1. `S -> N`
2. `C -> T`
3. `C -> I`
4. `I -> L`
5. `S, C, M -> G`
6. `S, M -> A`
7. `A -> M`

3 Obtain a lossless-join, BCNF decomposition of StudentInfo.

For FD (1): `S->N`, we decompose `SI` to
`S1 = (S N)`
`S2 = (S M A C T I L G)`

For FD (3): `I->L`, we decompose `S2` to
`S3 = (I L)`
`S4 = (S M A C T I G)`

For FD (2): `C->T,I`, we decompose `S4` to
`S5 = (C T I)`
`S6 = (S M A C G)`

For FD (4): `S,C,M->G`, we decompose `S6` to
`S7 = (S C M G)`
`S8 = (S C M A)`

For FD (6): `A->M`, we decompose `S8` to
`S9 = (A M)`
`S10 = (S C A)`

Note that FD (5) cannot be applied to any final decomposed relation, so the decomposition is done. FD (5): `S,M->A` is not preserved. So the final decomposition of StudentInfo is:
`S1 = (S N)`
`S3 = (I L)`
`S5 = (C T I)`
`S7 = (S C M G)`
`S9 = (A M)`
`S10 = (S C A)`

4 Obtain a lossless-join, dependency-preserving, 3NF decomposition of StudentInfo by making use of the BCNF decomposition in Question (c).

The BCNF decomposition we obtained from (c) does not preserve FD (5): `S,M->A`, so we add `S11 = (S M A)`.
So the final 3NF decomposition is:
`S1 = (S N)`, `S3 = (I L)`, `S5 = (C T I)`, `S7 = (S C M G)`, `S9 = (A M)`, `S10 = (S C A)`, `S11 = (S M A)`

Since `S9` $\subset$ `S11`, we can drop `S9`. So the final decomposition is:
`S1 = (S N)`
`S3 = (I L)`
`S5 = (C T I)`
`S7 = (S C M G)`
`S10 = (S C A)`
`S11 = (S M A)`

5 Obtain a lossless-join, dependency-preserving, 3NF decomposition of StudentInfo via synthesis by making use of your minimal cover in Question (b).

The minimum cover is `S->N, C->T, C->I, I->L, SCM->G, SM->A, A-M`.
The relations obtained from a minimum cover set is:
`R1 = (S N)`, `R2 = (C T)`, `R3 = (C I)`, `R4 = (I L)`, `R5 = (S C M G)`, `R6 = (S M A)`, `R7 = (A M)`

Check if any relation contains a key to the original relation: `R5 = (S C M G)` contains key `(S C M)`, so it is a lossless-join decomposition.

Since `R7` $\subset$ `R6`, we can drop `R7`. Since `R2` and `R3` share the same key `C`, we can merge them to `R23 = (C T I)`.
So the final decomposition is:

```
R1 = (S N)
R23 = (C T I)
R4 = (I L)
R5 = (S C M G)
R6 = (S M A)
```

6 Comment on the differences, if any, between your answers to Questions (d) and (e).

The decomposition obtained from (d) has one extra relation `R = (S C A)` comparing to the decomposition obtained from (e).