Tutorial: Third Normal Form

Your company, Apasaja Private Limited, is commissioned by Toko Kopi Luwak to design relational schema for the management of their coffee beans, drinks, and cafes.

A coffee bean is fully identified by its unique brand name or a combination of its cultivar and region (since the same cultivar can be grown in different region). For instance, we may have a coffee bean named "The Waterfall" which comes from a Tabi cultivar grown in Colombia.

A drink can be made utilizing a particular coffee bean. The name of the drink is only unique for a particular coffee bean. This means, we can have an "Espresso" made with "The Waterfall" or "La Bella" (which is a Pacamara cultivar grown in Guatemala). The price of the drink is also recorded.

A branch identified by its branch name may then sell the drink. A drink may be sold by zero or more branches. A branch may sell zero or more drinks. Additionally, the address of the branch is also recorded. Lastly, for each drink sold by a branch, we record the quantity sold to see which branch is the most profitable.

We are only given an abstract schema for this application as follows.

$$R = \{A, B, C, D, E, F, G, H\}$$

$$\Sigma = \{\ \{A\} \to \{C, E\},\ \ \{A, B\} \to \{D\},\ \ \{F\} \to \{H\},\ \ \{C, E\} \to \{A\},\ \ \{B, C, E\} \to \{D\},\ \ \{A, B, F\} \to \{D, G\},\ \ \{B, C, E, F\} \to \{G\}\ \}$$

This tutorial continues from the computation of candidate keys and minimal cover in "Tutorial: Functional Dependencies". You are advised to compute them before continuing. This tutorial also uses the schema from "Tutorial: Creating and Populating Tables" before the updates done during the tutorial.

Questions

Not all questions will be discussed during tutorial. You are expected to attempt them before coming to the tutorial. You may be randomly called to present your answer during tutorial. You are encouraged to discuss them on Canvas Discussion.

1. **Third Normal Form**.
   (a) Is R with Σ in 3NF?

   Comments:
   No.
   From "Tutorial: Boyce-Codd Normal Form", we know that $\{A\} \to \{C\}$ is non-trivial and $\{A\}$ is not a superkey. Unfortunately, $\{C\}$ is a prime attribute. So, we have to look for other violations.
   Consider $\{A, B\} \to \{D\}$.
   - It is non-trivial (since $\{D\} \not\subseteq \{A, B\}$).
   - $\{A, B\}$ is not a superkey (since it is not a superset of either key $\{A, B, F\}$ or $\{B, C, E, F\}$).
   - D is not a prime attribute.
     Prime attributes are $\{A, B, C, E, F\}$.

2. **Normalization**.
   (a) Decompose[1] R with Σ into a 3NF decomposition using the algorithm from the lecture.

   Comments:
   We can start from a canonical cover directly.
   $$\{\ \{A\} \to \{C, E\},\ \ \{F\} \to \{H\},\ \ \{C, E\} \to \{A\},\ \ \{B, C, E\} \to \{D\},\ \ \{B, C, E, F\} \to \{G\}\ \}$$
   For each functional dependency, we synthesize a fragment.
   $$\{\ \{A, C, E\},\ \{F, H\},\ \{A, C, E\},\ \{B, D, C, E\},\ \{B, C, E, F, G\}\ \}$$
   If there is any fragment that can be *subsumed*, we remove it from the result.
   $$\{\ \{A, C, E\},\ \{F, H\},\ \cancel{\{A, C, E\}},\ \{B, D, C, E\},\ \{B, C, E, F, G\}\ \}$$
   If none of the fragments contains a candidate key, we add one fragment for any key. Luckily, the key $\{B, C, E, F\}$ is a subset of $\{B, C, E, F, G\}$. So, we do not have to add another relation.
   $$\{\ \{A, C, E\},\ \{F, H\},\ \{B, D, C, E\},\ \{B, C, E, F, G\}\ \}$$

   (b) Is the result dependency preserving?

   Comments:
   Yes. This is guaranteed by the algorithm.

[1] Although it is a *synthesis* algorithm, the process is still decomposition.

Comments:

It is important to note that the BCNF and 3NF property does not guarantee anything by itself. In other words, we can find decompositions in BCNF or 3NF that are neither lossless-join nor dependency preserving. The benefit comes from the algorithm.

3. **3NF or BCNF**.

In this question, we will be revisiting the "issues" with the original schema from "Tutorial: Creating and Populating Tables". For simplicity, we will work with only the relevant attributes. The following tables provide the mapping from those attributes to letters.

`book` (A,B)

Attribute	Letter
`isbn`	A
`title`	B

`student` (C,D,E)

Attribute	Letter
`email`	C
`department`	D
`faculty`	E

`copy` (A,C,F)

Attribute	Letter
`copy`	F

`loan` (A,C,F,G,H)

Attribute	Letter
`borrowed`	G
`returned`	H

We also rename `owner` to `email` and `book` to `isbn`.

Note that some attributes are not listed with the table as they are referencing another table. We assume those attributes are present nonetheless.

You are advised to figure out the functional dependencies on your own. Check that they match the following set of functional dependencies.

$\Sigma = \{ \{A\} \to \{B\},\ \ \{C\} \to \{D, E\},\ \ \{A, C, F, G\} \to \{H\} \}$

(a) Recall that we found that `department` should uniquely identify `faculty`. This is represented as $\{D\} \to \{E\}$. Is the table `student` still in BCNF if we add $\{D\} \to \{E\}$ to Σ? Is it in 3NF?

Comments:

First we find the projection of Σ onto `student` table. The result is the following.

$$\{ \{C\} \to \{D, E\},\ \{D\} \to \{E\} \}$$

Next, we compute the candidate keys of `student` using this projection. The candidate key is $\{C\}$. Consider $\{D\} \to \{E\}$.

(i) It is non-trivial.

(ii) $\{D\}$ is not the superkey of `student`.

(iii) E is not a prime attribute.

Due to (i) and (ii), `student` is not in BCNF.
Due to (i), (ii), and (iii), `student` is not in 3NF.

This is the reason why we split the table and created a new table with only the attributes `department` and `faculty`. Verify that after the split, the tables are in BCNF.

(b) Consider instead that we replace $\{C\} \rightarrow \{D, E\}$ in Σ with $\{C, E\} \rightarrow \{D\}$ and $\{D\} \rightarrow \{C\}$. Is the table `student` still in BCNF? Is it in 3NF?

Comments:

First we find the projection of Σ onto `student` table. The result is the following.

$$\{\ \{C, E\} \rightarrow \{D\},\ \{D\} \rightarrow \{C\}\ \}$$

Next, we compute the candidate keys of `student` using this projection. The candidate keys are $\{C, E\}$ and $\{D, E\}$. The prime attributes are $\{C, D, E\}$.

(i) Consider $\{C, E\} \rightarrow \{D\}$. $\{C, E\}$ is a superkey as it is a superset of (in fact, it is) one of the candidate keys.

(ii) Consider $\{D\} \rightarrow \{C\}$. C is a prime attribute, but D is not a superkey. So this is a BCNF violation, but not a 3NF violation.

Observation. Since we found that all attributes are prime attributes, the table is always going to be in 3NF as there can be no violation.

Comments:

By reasoning via functional dependencies and normalization, we can refine our earlier schema designed using entity-relationship diagram. Both techniques complement each other.

References

[1] S. Bressan and B. Catania. *Introduction to Database Systems*. McGraw-Hill Education, 2006. ISBN: 9780071246507.

[2] Hector Garcia-Molina, Jeffrey D. Ullman, and Jennifer Widom. *Database Systems: The Complete Book*. 2nd ed. Prentice Hall Press, 2008. ISBN: 9780131873254.

[3] Raghu Ramakrishnan and Johannes Gehrke. *Database Management Systems*. 2nd. USA: McGraw-Hill, Inc., 2000. ISBN: 0072440422.