Database Design I: Functional Dependencies

DSC 301: Lecture 8

February 19, 2021

Lecture Objectives

- Functional dependencies

Introduction

We have been querying single table databases. However, the power of the relational database model is in its use of multiple relations (i.e., tables). In this lecture, we return to the relational data model and examine essential elements in its **design** structure. The quality of the design directly affect the ability to properly query the database. Poor design makes querying very difficult. Understanding design makes for more efficient MySQL programmer. The key (no pun intended) piece of theory is functional dependencies.

Functional Dependencies (FD)

- Tool to improve the design of relational model.
  - Describes relationship (constraint) among attributes
  - Used to determine keys
  - Used to **define normal forms -** decomposition of relation into two or more relations to remove anomalies.
  - Used to reason about queries and query optimization
  - Used in data storage and compression
    - e.g., eliminate redundancy
- FD is a constraint that **generalizes the concept of a key**.
  - A *key* is a special case of FD

Definition: Let A and B be sets of attributes[1] of relation R. A **functional dependency**, $A \rightarrow B$ exists if for every $x, y \in R$,

$$\pi_A(x) = \pi_A(y) \;\Rightarrow\; \pi_B(x) = \pi_B(y)$$

Definition (Equivalent): Let A and B be sets of attributes of relation R. Then there is a functional dependency $A \rightarrow B$ if the two tuples $t_i[B] = t_j[B]$ when $t_i[A] = t_j[A] \in R$ for all i, j.

- Notation and interpretation
  - The left hand side (A) is called the *determinant*
  - We say A "functionally" determines B, (or B is dependent on A)
  - Standard mathematical notation can also be used, $f(A) = B$.
  - Note: functional dependency on R is a constraint on the schema, not just a one-off property of a particular instance.
  - **Key dependencies** are a special case of functional dependencies. That is, if U is the set of all attributes of a relation, then a **key dependency** is a functional dependency of the form $K \rightarrow U$.
- **Example**: Table **??** shows a relation R with four attributes (A, B, C, D). We can see that $A \rightarrow B$ exists. What are some others?

Table 1: Example: Functional Dependency

R =

A	B	C	D
a_1	b_1	c_1	d_1
a_1	b_1	c_3	d_2
a_2	b_1	c_3	d_3
a_2	b_1	c_3	d_3
a_3	b_6	c_3	d_4

- **Full Functional Dependency**: $A \rightarrow B$, but if $C \subset A$, $C \not\rightarrow B$. That is, B is functionally dependent on A but not on a proper subset of A. Said another way, if removal of an attribute from the set A "breaks" the dependency. For example, in Table **??**, D is fully functional dependent on AC, but not on A.
  - If B depends on a subset of A, then we say that $A \rightarrow B$ is a **partial dependency**. In the case of keys, a partial dependency is a superkey.
- A functional dependency $A \rightarrow B$ is **trivial** if $B \subseteq A$.
  - i.e., $A \rightarrow A$ is a trivial FD, because $A \subseteq A$.

[1] Attribute A can represent a set of multiple attributes, i.e., $A = \{A_1, A_2, \ldots, A_m\}$, or a single set. We simply write A to represent this set. Similarly for B, C, ...

Armstrong Axioms and derived rules

Denote the sets of attributes as $A = \{A_1, A_2, \cdots, A_i\}$, $B = \{B_1, B_2, \ldots, B_j\}$, $C = \{C_1, C_2, \ldots, C_k\}$, and $D = \{D_1, D_2, \ldots, D_l\}$. Rules (1)–(3) are called **Armstrong Axioms** [**?**] and are used to derive **closure** of the set of functional dependencies. Rules (4)–(7) can be derived from (1)–(3). Armstrong's Axioms are *complete*.

1. **Reflexive**: if $B \subseteq A$, then $A \rightarrow B$.
2. **Augmentation**: If $A \rightarrow B$, then $AC \rightarrow BC$.
3. **Transitive dependence**: If $A \rightarrow B$ and $B \rightarrow C$, then $A \rightarrow C$.
4. **Self-determinant**: $A \rightarrow A$
5. **Decomposition**: If $A \rightarrow BC$, then $A \rightarrow B$ and $A \rightarrow C$.
6. **Union**: If $A \rightarrow B$ and $A \rightarrow C$, then $A \rightarrow BC$.
7. **Composition**: If $A \rightarrow B$ and $C \rightarrow D$, then $AC \rightarrow BD$.

Table 2: **Example**: Functional Dependency $A \rightarrow B$

R =

A	B	C	D
1	1	1	1
1	1	3	1
2	1	3	4
2	1	3	1
3	6	3	1

Goal: a minimal set of completely nontrivial FDs such that all FDs that hold on the relation follow from the dependencies in this set.