Normalization-1

Functional Dependencies

Seminar Session

(Monday, Sept. 21, 2015)

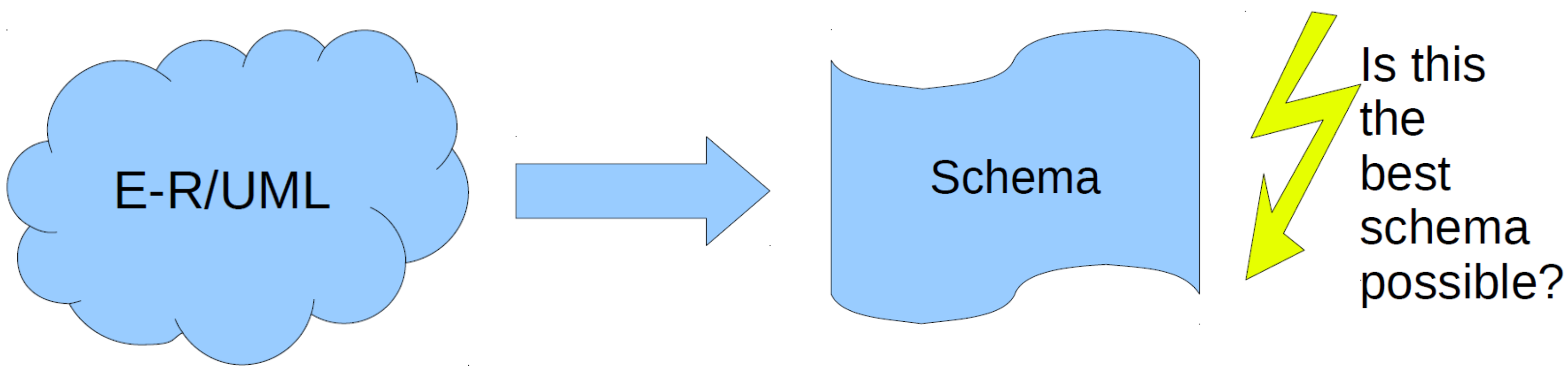
AGENDA

- Determine
  - Functional dependencies
  - Minimal basis
  - Closures
  - Keys of relations

Motivation

Why is knowing about Normal Forms important?

- Non-normalized data is tedious to write and maintain code for
- It is also more inefficient to query in SQL databases



Basic Concepts: Functional Dependencies (FDs)

- A → BC (A,B,C are attributes of a relation)
- When a tuple agrees with another one in the value of A, it must also agree on the values of B &C
- Not necessarily the other way around

A	B	C	D
1	2	3	5
4	2	3	6
5	2	5	7
5	2	5	8

Basic Concepts: Keys

- Given
  - If we have a relation with attributes $\mathbf{a_1, a_2, \ldots, a_n}$
- Superkey
  - the subset S of those attributes, where $\mathbf{S \rightarrow \{a_1, a_2, \ldots, a_m\}}$ is a superkey of the relation
  - Can you guess the superkeys?
    - Title, Year, StarName
    - Title, Year, StarName, Length, StudioName
    - …

title	*year*	*length*	*genre*	*studioName*	*starName*
Star Wars	1977	124	SciFi	Fox	Carrie Fisher
Star Wars	1977	124	SciFi	Fox	Mark Hamill
Star Wars	1977	124	SciFi	Fox	Harrison Ford
Gone With the Wind	1939	231	drama	MGM	Vivien Leigh
Wayne's World	1992	95	comedy	Paramount	Dana Carvey
Wayne's World	1992	95	comedy	Paramount	Mike Meyers

Basic Concepts: Closure

- Definition
  - A closure of a set of attributes $\mathbf{s=\{a_1,a_2..a_n\}}$ which participate in a set of functional dependencies ***F*** together with attributes $a_1, a_2,..., a_m$ (m>=n)
- Representation
  - Denoted as **s+**
- Semantics
  - The whole set of attributes that can be functionally determined by s (a superset of **s**, i.e., **s+**)

Basic Concepts: Closure…

- Example (***Combining***/Splitting)
  - A → B
  - B → C
  - The closure of A (A+) is ABC because
  - A → A (trivial)
  - A → B (given)
  - A → C (transitivity)
  - So A → ABC (composition)
- Example (Left Splitting)
  - **FD:** Title, Year → Length
  - Can we say Title → Length and Year → Length ?

Basic Concepts: Minimal Basis

- What is Minimal Basis?
  - Splitting 1 relation into 2 relations
  - Reduce the FDs
- Consideration
  - Must take care to preserve functional dependencies
- Possible Issues
  - If done wrong it can result in extra, meaningless tuples appearing in joins
- Related Question
  - Could it possibly result in less tuples appearing?

Conditions for Minimal Basis

1. All the FD's in B have singleton right sides.
2. If any FD is removed from B, the result is no longer a basis.
3. If for any FD in B we remove one or more attributes from the left side of F, the result is no longer a basis.

Inference Rules: Armstrong's Axioms

1. *Reflexivity.* If $\{B_1, B_2, \ldots, B_m\} \subseteq \{A_1, A_2, \ldots, A_n\}$, then $A_1A_2 \cdots A_n \rightarrow B_1B_2 \cdots B_m$. These are what we have called trivial FD's.

2. *Augmentation.* If $A_1A_2 \cdots A_n \rightarrow B_1B_2 \cdots B_m$, then

$$A_1A_2 \cdots A_nC_1C_2 \cdots C_k \rightarrow B_1B_2 \cdots B_mC_1C_2 \cdots C_k$$

for any set of attributes $C_1, C_2, \ldots, C_k$. Since some of the C's may also be A's or B's or both, we should eliminate from the left side duplicate attributes and do the same for the right side.

3. *Transitivity.* If

$$A_1A_2 \cdots A_n \rightarrow B_1B_2 \cdots B_m \text{ and } B_1B_2 \cdots B_m \rightarrow C_1C_2 \cdots C_k$$

then $A_1A_2 \cdots A_n \rightarrow C_1C_2 \cdots C_k$.

Exercise 1: FDs and Keys

a) **Order** (Product ID, Product Name, Customer ID, Customer Name, Date, Item Price, Amount, VAT, Gross Total, Net Total)

- All functional dependencies of Order?

Product ID	→ Product Name, Item Price, VAT
Customer ID	→ Customer Name
Product ID, Customer ID, Date	→ Amount
Item Price, Amount	→ Net Total
Net Total, VAT	→ Gross Total
Product Name	→ Product ID*
Customer Name	→ Customer ID*

Exercise 1: FDs and Keys…

a) **Order** (Product ID, Product Name, Customer ID, Customer Name, Date, Item Price, Amount, VAT, Gross Total, Net Total)

- What are the key candidates?

{Product ID, Customer ID, Date}

If product/customer names are unique, also

{Product ID, Customer Name, Date}
{Product Name, Customer ID, Date}
{Product Name, Customer Name, Date}

Exercise 1: FDs and Keys…

b) **Student** (StudentID, Address, Course, TA)

- Functional dependencies?
  - StudentID

    StudentID → Address
  - StudentID, Course

    StudentID, Course → TA
- Which field(s) are the superkeys of the relation?

  The set of all attributes is a trivial superkey.
  StudentID, Course is a candidate (minimal) key.

Exercise 2: Closure

a) Given a relation R (A,B,C,D,E,G) with *F*

$$AB \rightarrow C,\ BC \rightarrow D,\ D \rightarrow EG,\ CG \rightarrow BD$$
$$C \rightarrow A,\ ACD \rightarrow B,\ BE \rightarrow C,\ CE \rightarrow AG$$

We define α = BD.

- Find the closure of attributes **α+** of (F, α).

No.	Results	Func. Depend. Used
1	BD	D → EG
2	BDEG	BE → C
3	BCDEG	CG → BD
4	no change	CE → AG
5	ABCDEG	AB → C
6	no change	C → A
7	no change	BC → D
8	no change	ACD → B
9	no change	D → EG
10	no change	BE → C
11	no change	CG → BD
12	no change	CE → AG

Exercise 2: Closure…

b) Given a relation R (A,B,C,D,E,F,G) with *F*:

$\boldsymbol{A \rightarrow BC,\ E \rightarrow CF,\ B \rightarrow E,\ CD \rightarrow EF,\ A \rightarrow G}$

- Find the closure of AB.

A → B (because A → BC)
A → E (because A → B and B → E)
A → CF (because A → E and E → CF)
A → G
Therefore, A → ABECFG => AB → ABECFG

Exercise 2: Closure…

b) Given a relation R (A,B,C,D,E,F,G) with *F*:

$$\boldsymbol{A \to BC,\ E \to CF,\ B \to E,\ CD \to EF,\ A \to G}$$

- Key of relation?

  AD → ABCDEFG [*so AD is a key*]

- Closure of G?

  Just G

Exercise 3: Minimal Basis

a) Given

$$A \rightarrow BC,\ B \rightarrow C,\ A \rightarrow B,\ AB \rightarrow C$$

- Find the minimal basis?
- Solution:
  - AB→C
    - reduced to A→C because {C}⊆ Closure(F, A) = (A), (AB), (ABC)
  - A→BC
    - is implied by A→C, A→B
  - A→C
    - is implied by A→B, B→C
  - The only functional dependencies we actually need are **A→B, B→C**

Exercise 3: Minimal Basis…

b) Given:

$$AB \rightarrow C,\ C \rightarrow A,\ BC \rightarrow D,\ ACD \rightarrow B$$
$$BE \rightarrow C,\ CE \rightarrow FA,\ CF \rightarrow BD,\ D \rightarrow EF$$

- Find the minimal basis?
  - Reduction of the left side: ***ACD → B***
    - ***ACD → B*** can be reduced to *CD → B* because {*B*} ⊆ *Closure*(*F, CD*) = (*CD*), (*CDEF*), (*CDEFB*)
  - Reduction of the right side: ***CD → B, CE → FA, CF → BD***
    - ***CD → B*** can be reduced to *CD* → 0 because {*B*} ⊆ *Closure*(*F* \{*CD* → *B*}, *CD*) = (*CD*), (*CDEF*), (*CDEFB*)
    - ***CE → FA*** can be reduced to *CE → F* because {*A*} ⊆ *Closure*(*F* \{*CE → FA*} + {*CE → F* }, *CE*) = (*CE*), (*CEF*), (*CEFA*)
    - ***CF → BD*** can be reduced to *CF → B* because {*D*} ⊆ *Closure*(*F* \{*CF → BD*} + {*CF → B*}, *CF*) = (*CF*), (*CFB*), (*CFBD*)
  - As minimal basis remains:
    - *AB → C, C → A, BC → D, BE → C, CE → F , CF → B, D → EF*
  - An alternative solution would be to right-reduce *CF → BD* instead of *CD → B*

Exercise 3: Minimal Basis…

Projecting a set of functional dependencies

c) Given: R(A, B, C, D), FD's: $A \rightarrow B, B \rightarrow C, C \rightarrow D$

- We wish to project out attribute B, leaving a relation R1(A, C, D), Find the FD's for R1?
- We need to take closure of all eight subsets of {A, C, D} using the full set of FD's
- Simplifications
  - Closing the empty set and the set of all attributes cannot yield a non trivial FD
  - If we already know that the closure of some set X is all attributes, then we can not discover any new FD's by closing superset of X

Exercise 3: Minimal Basis…

Projecting a set of functional dependencies…

c) Given: R(A, B, C, D), FD's: $A \rightarrow B, B \rightarrow C, C \rightarrow D$

- We wish to project out attribute B, leaving a relation R1(A, C, D), Find the FD's for R1?
- Lets find closures of singleton sets
  - {A}+ = {A, B, C, D}
    - FD's: A → C, A→ D holds in R1 and **A → B** in R
  - {C}+ = {C, D}
    - FD: C → D
  - {D}+ = {D}
- Doubleton sets
  - { … {A}
  - { … al FD

Can we further reduce it?

Hint: Transitivity

- Resultant FD's: A→C, A→D, C→D

Projecting a set of functional dependencies: Algorithm

INPUT: A relation R and a second relation R_1 computed by the projection $R_1 = \pi_L(R)$. Also, a set of FD's S that hold in R.

OUTPUT: The set of FD's that hold in R_1.

METHOD:

1. Let T be the eventual output set of FD's. Initially, T is empty.

2. For each set of attributes X that is a subset of the attributes of R_1, compute X^+. This computation is performed with respect to the set of FD's S, and may involve attributes that are in the schema of R but not R_1. Add to T all nontrivial FD's $X \rightarrow A$ such that A is both in X^+ and an attribute of R_1.

3. Now, T is a basis for the FD's that hold in R_1, but may not be a minimal basis. We may construct a minimal basis by modifying T as follows:

   (a) If there is an FD F in T that follows from the other FD's in T, remove F from T.

   (b) Let $Y \rightarrow B$ be an FD in T, with at least two attributes in Y, and let Z be Y with one of its attributes removed. If $Z \rightarrow B$ follows from the FD's in T (including $Y \rightarrow B$), then replace $Y \rightarrow B$ by $Z \rightarrow B$.

   (c) Repeat the above steps in all possible ways until no more changes to T can be made.

Assignment #6

- You need to answer queries related to
  - SQL query results
  - Functional Dependencies
- Detail description is available at [**Link**]
  - You can also find it in General folder at Google drive

Good Luck...