CS 273 Homework

Shiyu Ji

1 Problem A

Your task is to design a database for Amazon that sells books. The following is the description of the application.

- Each book has a name and a publisher.
- Each book has many versions. Each version has a version number, the year and date it was first published, and the authors.
- Registered viewers can rate any version of a book. For each rate, we want to record its post date and rate.
- For each registered viewer, we want to record his/her userID and password.

If you do not have sufficient information, please state your assumptions clearly.

(a) Draw an ER diagram for this application. Be sure to mark the multiplicity of each relationship of the diagram. Decide the key attributes and identify them on the diagram.

(b) Convert the above ER diagram into a relational schema. Merge relations where appropriate. Your solution should have as few relations as possible, but they do not need to be normalized. Specify they key of each relation in your schema.

(c) Write a 3NF for the derived relational schema.

2 Problem B

The following questions refer to the database schema below: Product(pid, price, color), Order(cid, pid, quantity), Customer(cid, name, age).

(a) Write a query, in Relational Algebra, to return the names of customers who order at least one product with color "Yellow".

Answer:

$$\pi_{\texttt{name}}(\texttt{Customer} \bowtie \texttt{Order} \bowtie \sigma_{\texttt{color=Yellow}}(\texttt{Product})).$$

(b) Write a query in Tuple Relational Calculus and Domain Relational Calculus, to return the product with the highest price.

Answer:

- Tuple Relational Calculus:

$$\{t | t \in \texttt{Product} \wedge \forall s \in \texttt{Product}(s[\texttt{price}] \leq t[\texttt{price}])\}.$$

- Domain Relational Calculus:

$$\{< i, p, c > | < i, p, c > \in \texttt{Product} \wedge \forall < i', p', c' > \in \texttt{Product}(p' \leq p)\}.$$

(c) Write an SQL query, to return the total quantity of products ordered by customers with age greater than 50.

Answer:

```
select SUM(quantity)
from Order
where cid in (
    select cid
    from Customer
    where age > 50
);
```

(d) Write an SQL query, to return the pid(s) of the most ordered product(s) (i.e. the product(s) with the highest total ordered quantities).

Answer:

```
select pid
from Product
where pid in (
  select pid
  from (
    select pid, SUM(quantity) as QuantitySum
    from Order
    group by pid
  )
  having SUM(quantity) in (
    select MAX(QuantitySum)
    from (
      select Sum(quantity) as QuantitySum
      from Order
      group by pid
    )
  )
);
```

3 Problem C

We have an employee database with three tables. The Employee table stores information about employees. Every employee is identified by an **EmployeeID**. The Department table stores information about each department. The Vacations table stores information about the total number of days each employee takes for vacation.

- Employee(EmployeeID, FirstName, LastName, Office, Email, DepartmentID)
- Department(DepartmentID, DepartmentName)
- Vacations(EmployeeID, Days)

Answer the following queries using SQL and write its corresponding TRC if exists.
a) List the Office and Email of employee "John Smith".
Answer:

- SQL:

```
select Office, Email
from Employee
where FirstName = 'John' and LastName = 'Smith';
```

- TRC:

$$\{t|\exists s \in \texttt{Employee}(s[\texttt{Office}] = t[\texttt{Office}] \wedge s[\texttt{Email}] = s[\texttt{Email}] \wedge s[\texttt{FirstName}] = \texttt{'John'} \wedge s[\texttt{LastName}] = \texttt{'Smith'})\}.$$

b) List the number of vacation Days taken by each employee with the name "John Smith".
Answer:

- SQL

```
select Days
from Vacations
where EmployeeID in (
  select EmployeeID
  from Employee
  where FirstName = 'John' and LastName = 'Smith'
);
```

- TRC:

$$\{t|\exists v \in \texttt{Vacations}(v[\texttt{Days}] = t[\texttt{Days}] \wedge \exists e \in \texttt{Employee}(e[\texttt{EmployeeID}] = v[\texttt{EmployeeID}] \wedge e[\texttt{FirstName}] = \texttt{'John'} \wedge e[\texttt{LastName}] = \texttt{'Smith'}))\}.$$

c) List the FirstName and LastName of all employees who never took a vacation day.
Answer:

- SQL:

```
select FirstName, LastName
from Employee
where EmployeeID in (
  select EmployeeID
  from Vacations
  where Days = 0;
);
```

- TRC:

$$\{t|\exists s \in \texttt{Employee}(t[\texttt{FirstName}] = s[\texttt{FirstName}] \wedge t[\texttt{LastName}] = s[\texttt{LastName}] \wedge \exists v \in \texttt{Vacations}(v[\texttt{EmployeeID}] = s[\texttt{EmployeeID}] \wedge v[\texttt{Days}] = 0))\}.$$

d) List the DepartmentName of every department whose total number of vacation days taken by its employees is the largest among those of all the departments.
Answer:

```
select DepartmentName
from Department
where DepartmentID in (
  select DepartmentID
  from (
```

```
    select DepartmentID, SUM(Days) as DaySum
    from Employee inner join Vacations
    on Employee.EmployeeID = Vacations.EmployeeID
    group by DepartmentID
  )
  having SUM(Days) in (
    select MAX(DaySum)
    from (
      select SUM(Days) as DaySum
      from Employee inner join Vacations
      on Employee.EmployeeID = Vacations.EmployeeID
      group by DepartmentID
    )
  )
);
```

4 Problem D

We perform decomposition to normalize an original schema to be of certain normal forms. For such a decomposition to be "equivalent" to the original schema, it is desirable to be lossless. To study this concept, let us consider an original schema $R(A, B, C)$. Suppose we decompose R into $R_1(A, B)$ and $R_2(B, C)$.

(a) Is this decomposition always lossless? Answer yes or no and briefly explain why.

Answer: No. If there is a functional dependency $A \to BC$, then this dependency will be lost by such a decomposition. In particular, suppose two tuples have the same value of B but different values of A and C. Then the decomposed tuples in R_1 and R_2 give 4 possible combinations, implying it is impossible to recover the original schema.

(b) Give an example instance of R (i.e. an example table with several tuples) and demonstrate its decomposition, to support your answer in (a).

Answer: A "lossy" relation R can given as follows:

A	B	C
3	0	0
4	0	1

After decomposition R_1 contains $(3, 0)$ and $(4, 0)$, and R_2 contains $(0, 0)$ and $(0, 1)$. Clearly the possible schema before the decomposition has another possibility: $(3, 0, 1)$ and $(4, 0, 0)$, implying it is impossible to recover the original schema.

5 Problem E

Given below is the set **F** of functional dependencies for the relational schema:

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

$$A \to BC$$

$$BD \to E$$

$$E \to F$$

$$F \to D$$

$$E \to D$$

1. Is the set **F** a minimal cover? Explain the reason. If **F** is not minimal cover, find a minimal cover for **F**.

Answer: No, **F** is not a minimal cover, since the dependency $E \to D$, which can be inferred by $E \to F$ and $F \to D$, is redundant.

By removing the redundant dependency we have the minimal cover as follows:

$$A \to BC$$

$$BD \to E$$

$$E \to F$$

$$F \to D$$

2. Based on the minimal cover for **F**, produce a lossless BCNF decomposition for this schema. Is the result dependency-preserving? Explain why or why not.

Answer: A lossless BCNF decomposition can be

$$R_1(\underline{A}, B, C)$$

$$R_2(\underline{E}, F)$$

$$R_3(\underline{F}, D)$$

$$R_4(\underline{A, E})$$

The BCNF above does not preserve the dependency $BD \to E$.

3. Based on the minimal cover for **F**, produce a dependency preserving 3NF decomposition. Is the result redundancy free? Explain why or why not.

Answer: A 3-NF decomposition can be

$$R_1(\underline{A}, B, C)$$

$$R_2(\underline{B, D}, E)$$

$$R_3(\underline{E}, F)$$

$$R_4(\underline{F}, D)$$

It contains redundancy of attribute D, which appears in both R_2 and R_4.

6 Problem F

1. Consider the relation Treatment and FDs below. Describe, with examples, two potential issues that can arise with this design.

- Treatment(doctorID, doctorName, patientID, diagnosis)
- doctorID $\to$ doctorName
- doctorID, patientID $\to$ diagnosis

Answer:

- A doctor may have many patients (e.g., 100 per month), then there are many repeated doctorName in the table, giving too much redundancy.
- If one needs to update doctorName, it can easily give inconsistency to this table. If in the table of doctors, the doctorName is changed, then every doctorName in this table has to be updated as well.

2. Prove that every two-attribute relation is in BCNF.

Proof. For any two-attribute relation $R(A, B)$, we have

- If without loss of generality A is the primary key, then $A \to B$ is the only non-trivial dependency.
- If A and B together form the primary key, then every dependency is trivial.

For either case the condition of BCNF is satisfied. □

3. Prove that if relation R is in 3NF and every key is simple (i.e., a single attribute), then R is in BCNF.

Proof. Without loss of generality in any 3NF $R(A_0, \cdots, A_n)$ every non-trivial dependency $A_i \cdots A_j \to A_k$, we have two cases:

- $A_i \cdots A_j$ is a superkey, or
- A_k is in some key. Since every key in R is simple, A_k must be a key and thus a superkey.

Either case preserves the condition of BCNF. □

7 Problem H

There are two common methodologies (not specific techniques) to learn compact and meaningful word embeddings. List them and briefly describe the main idea.

Answer:

- Global co-occurrence statistics. The idea is to count the co-occurrences throughout the full documents, i.e., to compute the word-doc occurrence matrix for all the words in the same document.
- Prediction in local windows. The idea is to count the co-occurrences in a local window, which contains the words within a certain distance.

8 Problem I

(a) Given a query q and a set of documents d_1, d_2, $\cdots$, d_m, use the vector space model to show how to rank documents. Assume q and d_i are sets of words.

(b) Describe the intuition behind text length normalization and write down a normalization formula.

(c) Describe the intuition behind language model smoothing and write down a smoothing function.

9 Problem J

(a) Write down the objective function of K-means.

(b) Assume you have n d-dimension vectors, write down the code of K-means to cluster these n vectors to K groups.

(c) Explain three methods to measure the distance between two clusters for numerical data.

10 Problem K

Explain the connection and difference between two link-based ranking models: HITS and PAGERANK, and write down their formula.

11 Problem L

In order to build a knowledge graph from a large text repository, we need to perform several tasks including entity extraction. Could you please describe these tasks and their connections?

12 Problem M

Describe the basic steps of RandomForest.

13 Problem N

Explain how to cluster the following dataset.

14 Problem O (Bonus Problem)

Assume Facebook uses relational database technology to record its user information and user friendship information. It has two large tables (UserId, Gender), (USerId1, UserId2).

1. Writing a SQL query to list all of distinct 2-hop female friends of user "Mary".
2. Writing a SQL query to find the user who has the maximal number of 2-hop friends.
3. Propose a query that could not be easily expressed by SQL. Please explain why.