Data Quality: State of the art

Data Quality?

- Data Quality (DQ) is a perception of data’s fitness to serve its purpose in a given context.
- DQ problem costs U.S. businesses around $600 billion annually ([TDWI Journal])
- DQ is characterized by [no silver bullet for DQ problems]: Dimensions: Correctness, Consistency, Accuracy, Timeliness...

Data Completeness?

- Is all necessary data present?
- Measure: The extent to which data are of sufficient breadth, depth, and scope for the task at hand, i.e. query answering.

Motivation Examples

Schema

pupil(name, level, code)  ... pupil
class(level, code, dept) ... every class belongs to a department
langAtt(name, language) ... pupils attend language courses

Plain reasoning

TC Statement 1: We are complete for all pupils.
TABLE: pupil(name,level,Class) WHERE:
TC Statement 2: We are complete for all pupils in the class 1a.
pupil(name,Level,Class) WHERE: Level=1 AND Class='a'

Query 1: Who are the pupils at the 1st class?
SELECT p.name FROM pupil AS p WHERE p.level=1'

Can we answer Query 1 completely under the assumption of Statement 1?

Can we answer Query 1 completely under the assumption of Statement 2?

Reasoning under Finite Domains (FD)

FD 1: Codes of the pupils classes can be either a or b.
pupil(code). IN (a,b)
TC Statement 2: We are complete for all pupils in the class 1a.
TABLE: pupil(name,Level,Class) WHERE: Level=1 AND Class='a'

Can we answer Query 1 completely under the assumption of Statement 2 and FD1?

Is Query 1 complete if we in addition consider the TC-statement proposed by MAGIK?

Problem Statement

Given an information (meta-statements, called Table Completeness (TC) statements) that some parts of available database (D') is complete can we guarantee (deduce) that a query answer is the same (called Query Completeness (QC)) as the query is evaluated over the complete (ideal) database (D)?

- To express partial completeness of database we use table (local) completeness (TC) statements [H.Levy ’96]

TCi: We are complete for science pupils
- TABLE: pupil(Name,Level,Code)
WHERE: classLevel,Code,science)
pupil(N,L,C) = pupil(N,L,C) / class(l,C,science)

Let D1 = {class(1,a,sci), D2 = {class(1,a,sci)} and D3 = {class(1,a,sci), pupil(john,1,a)}

- (D',D2) satisfies TC1
- (D',D3) doesn't satisfy TC1
- Similarly for a query Q(A) = pupil(N,L,C)
Q is complete under (D',D2)
Q is NOT complete under (D',D3)

Goal: Automate the reasoning on query completeness (QC) given information about complete parts of a database (TC-statements).

Implementation

Encoding of the Problem in Logic Programming (Answer Set Programming

Finite Domains (FD)

Finite Domains (FD)

System Architecture

Implementation

Summary

- The first realized system that can reason about reasons on query completeness based on the partial database (table) completeness (TC-QC)
- It’s gone beyond original TC-QC problem, and we investigate the impact of Schema Constraints, like Foreign keys and Finite Domains, on TC-QC entailment.
- We developed a component for explanations and suggestions, in the case that the query is not complete, that indicates which parts of a database are incomplete w.r.t. the query.