

Data Civilizer

by

A Collection of Folks at MIT, QCRI,
Waterloo and TU Berlin

The Problem

- Mark Schreiber (Merck) reports that his data scientists spend 98% of their time
 - Locating data of interest
 - Accessing data of interest
 - Cleaning and transforming data of interest
- I.e. 39 hours a week of “mung work” and 1 hour a week doing the job for which they were hired
- NOBODY reports less than 80% mung work!

Data Civilizer

- Goal is to make Mark Schreiber happy
 - i.e. drive down the 98%

Data Civilizer

- Enterprise crawling to enable next steps
- Data Discovery
 - Find tables of interest to a data scientist
- Transformations
 - Syntactic (e.g. European dates to US dates)
 - Semantic (e.g. Merck has five different ID systems for chemical compounds)
- Join path identification and choice
- Data cleaning

Our Demo

- Enterprise crawling to enable next steps
- Data Discovery
 - Find tables of interest to a data scientist
- Transformations
 - Syntactic (e.g. European dates to US dates)
 - Semantic (e.g. Merck has five different ID systems for chemical compounds)
- Join path identification and choice
- Data cleaning

Context

- Merck has ~4000 Oracle data bases
- Plus a data lake
- Plus untold files
- Plus untold spreadsheets
- Plus they are interested in public data from the web
- Any solution has to work at scale!!!!!!

We Can't Do a Merck Demo

- They are protective of their data
 - We haven't cracked the problem of getting access to much of their data
- Ergo we don't have a suitable crawler

Instead.....

- We are using the MIT Data Warehouse
 - 2400 tables in an Oracle database
 - Students, courses, buildings, ...
 - 160 are “semi-public”
- Campus personal have ad-hoc questions
- For example:
 - How many employees work in degree granting departments?

**Analysts spend more time finding relevant data
than analyzing it**

Data Civilizer Discovery Module

- Goal: Find data relevant to the question at hand

Data Civilizer Discovery Module

- Goal: Find data relevant to the question at hand
- Challenge: **scale** and **varied discovery needs**

Data Civilizer Discovery Module

- Goal: Find data relevant to the question at hand
- Challenge: **scale** and **varied discovery needs**
- Approach to large scale data discovery:
 - **Data Summarization**
 - **Mining relationships: Linkage graph**
 - **Discovery algebra: express different queries**

Demo

Which Join Path is the Best?

- Each join path leads to a different view
 - different size - **coverage**
 - different quality - **cleanliness**
- Combine the two metrics to pick the path
- But, how to estimate cleanliness?

Estimating cleanliness

- Estimate the cleanliness of source data
 - . Outlier detection
 - . Check integrity constraints
 - . New method based on relationships in linkage graph
- Propagate cleanliness from source to view

View Cleaning with a Budget

- Where to clean
 - . Clean sources may waste budget on irrelevant cells
 - . Clean view may waste budget on duplicates
 - . Only clean source cells that affect the view
- Which cell to clean?
 - . Clean cells with the biggest impact to the view.
 - . Leverage cleanliness propagation to calculate the impact

Demo

What's Coming

- Eye Candy!!!!
- Semantic transformations
 - Using Data Xformer (CIDR 2015, SIGMOD 2015)
 - Inside the firewall as well as out on the web
- Partner to get syntactic ones
- Workflow system
 - Data Civilizer has to be iterative

What's Coming

- Join path clustering
 - To identify ones with the same semantics
 - Will require human input!
- Data cleaning cannot be totally manual
 - QCRI has done a lot of work in this area
 - We have a bunch of ideas on how to move forward
- Provenance
 - Mark is interested in what is derived from what

What's Coming

- Cannot copy all data of interest into a data lake
 - There is simply too much of it
- Have to access data “in situ” and on demand
 - Requires a polystore
 - And we have built one (BigDAWG)

Stay Tuned for a Complete System