CMSC 828E: Introduction

Amol Deshpande

University of Maryland, College Park

August 30, 2010

Today

- Overview
  - Grading etc. . .
  - Motivation
  - What this class is about, What we will cover

Today

- Overview
  - Grading etc. . .
  - Motivation
  - What this class is about, What we will cover
- Course Website
- Course Forum
  - "Register" → Go to "User CP" → join group "CMSC 828E Fall 2010"
  - We will use this for news, discussions, and for *posting paper critiques*

Today

- Overview
  - Grading etc. . .
  - Motivation
  - What this class is about, What we will cover
- Course Website
- Course Forum
  - "Register" → Go to "User CP" → join group "CMSC 828E Fall 2010"
  - We will use this for news, discussions, and for *posting paper critiques*
- Couple of things
  - No laptops
  - Typically won't use detailed slides

Overview

- We will cover:
  - Probabilistic Databases
  - Graph Databases
  - Large-scale, Cloud-based Analytics (somewhat)
- Prerequisite: CMSC 424
  - Some familiarity with databases is required
  - Class notes of 424 and 724 off of my webpage

Overview

- We will cover:
  - Probabilistic Databases
  - Graph Databases
  - Large-scale, Cloud-based Analytics (somewhat)
- Prerequisite: CMSC 424
  - Some familiarity with databases is required
  - Class notes of 424 and 724 off of my webpage
- Grading:
  - A class project (40%)
  - Take-home Exams (40%)
  - Paper critiques + class participation (20%)
    - Critiques mandatory before the class

Motivation

- Why are these problems interesting ?

Relational Databases: Overview

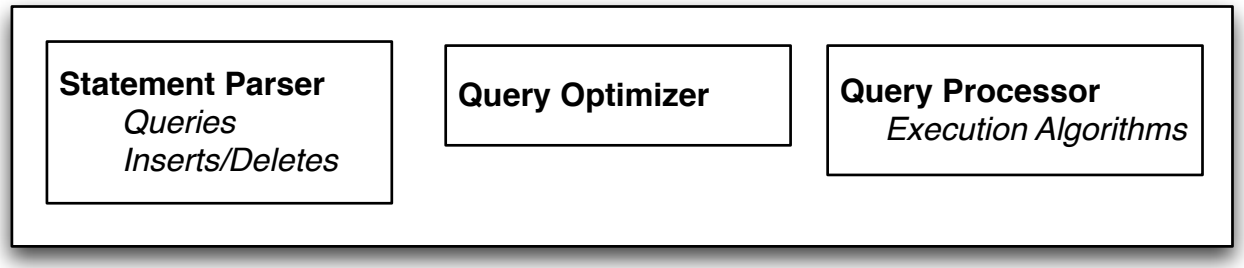
- Focusing on storage and query processing aspects
  - Ignoring “transactions”

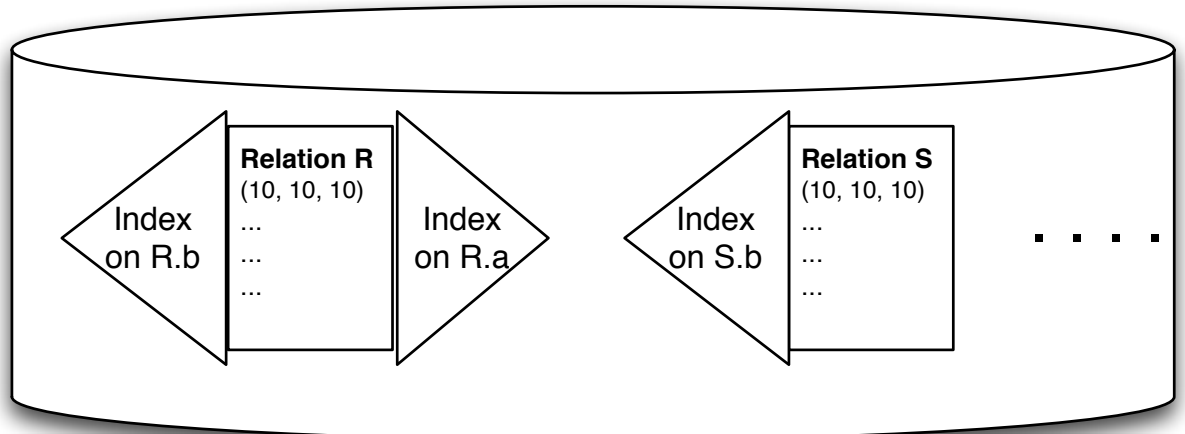
Query 1: select *
from R, S, T
where R.b = S.b and S.c = T.c

Query 2: select sum(a) as suma
from R
where R.b = 60

Other types of queries: Ranking/Top-k, Nearest Neighbor, Keyword Search etc...

Other types of commands: Create tables, insert tuples, delete tuples etc...





Relational Databases: Overview

- Some key advantages of RDBMS
  - Clean data model: tables and operations on tables that return tables
  - "Data Independence": user doesn't have to worry about how the data is stored, where
  - Declarative interfaces
    - SQL query language: easy to understand, low learning and usage curve
    - Visual interfaces
  - Nice abstractions that enable very good engineering
    - Modular structure (at least the query processor)
    - Each operation heavily optimized
    - Query optimization technology a great success
    - Indexing techniques developed for many data and query types

Motivation 1: Uncertainty

- Relational databases assume “exact” data
- Increasingly breaking down
- “Sensor” data
  - Automated sensing devices are becoming ubiquitous
  - e.g., environment monitoring sensors, GPS, RFID, vital signs (BP, heart rate)...
  - Often have errors for various reasons
    - Communication failures
    - Hardware limitations (can’t measure things precisely)
    - Software or hardware failures
  - Often much error correction on the device itself
  - But the end result still has noise/uncertainty

Motivation 1: Uncertainty

- Relational databases assume “exact” data
- Increasingly breaking down
- Data integration
  - Task of combining data from multiple sources
  - Many sources of uncertainties and errors
    - Would like to model trust and reputation of the sources
    - Often the “mapping” is done in an automated fashion
  - See this chapter for more details
- Information extraction
  - Task of automatically extracting structured information from natural language text
  - Extraction process rarely precise
  - e.g., 5-2703 can be a phone number, or an account number

Motivation 1: Uncertainty

- Relational databases assume “exact” data
- Increasingly breaking down
- Increasing use of machine learning techniques and predictive models
  - Many classifiers provide probabilities of belonging to different classes
  - May use predictive models for, say, stock market
- Scientific datasets
  - Experimental data or observation data often imprecise (at best, incomplete)
- Data may be exact, but queries may not be
  - inst-name like “UMD” (does this match “Univ of MD” ?)
  - Common in Information Retrieval

Types of uncertainty

Time	Source	News	Reputation/ Trust
3pm	Blog1	Accident on Route..	20%

Possibly stale data,
from untrusted sources

Time	Name	Confidence
2pm	John S. **or** Bob R.	40% (JS) 50% (BR)
2pm	Jane S.	40% (80% if above is John S.)

Probabilistic data
w/ complex correlations

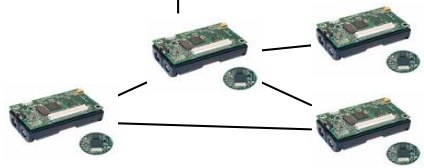
Sensor Data Fusion
and Aggregation

Time	Lane	Speed
10am	1	50±5 mph
10am	2	40±2 mph

Automatically extracted
imprecise data

Time	(X, Y)	Temp	Confidence
2pm	(20,20)	20±0.5	99%
2:10pm	(20, 20)	22±1.5	20%

Data with confidence and
accuracy bounds



Caption: Even if the sensor web data sources were to publish data using intuitive well-defined interfaces, the complex and semantically disparate measures of data quality and uncertainty typically associated with it make sensor data fusion and aggregation a challenging task.

Types of uncertainty

- “Nulls” (very interesting early work on this)
- More generally, missing values/incomplete databases

Types of uncertainty

- “Nulls” (very interesting early work on this)
- More generally, missing values/incomplete databases
- Existence uncertainty: Tuples may or may not exist
  - Can be modeled logically, probabilities, fuzzy logic etc. . .

Types of uncertainty

- “Nulls” (very interesting early work on this)
- More generally, missing values/incomplete databases
- Existence uncertainty: Tuples may or may not exist
  - Can be modeled logically, probabilities, fuzzy logic etc. . .
- Attribute values are uncertain
  - I saw a car, but not sure of the color
  - Sensor values have “noise” in them

Types of uncertainty

- “Nulls” (very interesting early work on this)
- More generally, missing values/incomplete databases
- Existence uncertainty: Tuples may or may not exist
  - Can be modeled logically, probabilities, fuzzy logic etc. . .
- Attribute values are uncertain
  - I saw a car, but not sure of the color
  - Sensor values have “noise” in them
- Approximations/Summarization
- Confidence/Trust/Quality
- “Correlations”
  - Either this tuple exists or that
  - If this attribute is 1, then that attribute is 2

How to deal with Uncertainty ?

- Approach 1: “Clean” the data
  - Remove noise; Thresholding (e.g., take majority opinion)
  - More up-front work, but standard DB operations afterwards
- Approach 2: Keep the uncertainty information and reason with it
  - Less up-front work
  - Query processing is much harder (next class)
- Why do Approach 2?
  - Smoother results; more flexibility; better decisions
    - Throwing away information can never be a good idea
  - Can corroborate information from multiple sources
- We mostly process information in that manner.

New types of tasks

- We may want to do more than SQL queries on uncertain data
- Inference types of queries
  - “If my prices were 10% higher, what would have been my profit last year?”
  - See the very interesting work on the MCDB project

New types of tasks

- We may want to do more than SQL queries on uncertain data
- Inference types of queries
  - “If my prices were 10% higher, what would have been my profit last year?”
  - See the very interesting work on the MCDB project
- Active Learning
  - Try to resolve uncertainty as you go along by asking questions, getting help from an expert

New types of tasks

- We may want to do more than SQL queries on uncertain data
- Inference types of queries
  - “If my prices were 10% higher, what would have been my profit last year?”
  - See the very interesting work on the MCDB project
- Active Learning
  - Try to resolve uncertainty as you go along by asking questions, getting help from an expert
- Budgeted queries
  - Given a budget for resolving uncertainty, do the best you can for a query
  - Some deep recent work on this and related problems

Motivation 2: Graph Data Management

- Graph-structured data is everywhere
  - Web; Social networks
  - Semi-structured data (XML)
  - Computer Networks
  - Biological data (protein-protein interaction networks); Chemical data analysis

Why relational databases won't work ?

- Have two tables: one for nodes, one for edges
  - More if there are different types of nodes or edges
  - Doesn't work

Why relational databases won't work ?

- Have two tables: one for nodes, one for edges
  - More if there are different types of nodes or edges
  - Doesn't work
- Queries and tasks do not look like SQL at all
  - Subgraph pattern queries (also, approximate searching)
  - Keyword search; Proximity search; Reachability queries
  - "Count the number of triangles?"
    - Computation of several interesting statistics requires that
  - Do "entity resolution"
    - i.e., decide if two nodes refer to the same entity
    - e.g., authors in citation networks; email addresses in social networks
  - Compute "PageRank"
    - e.g., to decide important nodes
  - How does an outbreak spread in this graph ?

Why relational databases won't work ?

- Storage and indexing techniques must be reconsidered
  - Web graphs and social network graphs can be highly compressed using specialized techniques
  - New indexing techniques needed
  - Different considerations if we have:
    - one very large graph (> 100 million nodes)
    - vs millions of small graphs
- New declarative interfaces needed
  - e.g., for specifying complex tasks like entity resolution
- Such networks are often highly dynamic
  - How to do "continuous" query processing over such data

Some of the challenges we will focus on

- Uncertainty
  - Edges may or may not exist
  - Probability distributions over attribute values
  - Two nodes may be identical with a probability (not considered before)
- Temporal and dynamics issues
  - An edge may exist for a given time duration
  - Rapidly changing graphs
    - E.g. how to do pattern detection in presence of streaming graph data ?

Motivation 3: Cloud Computing

- Data is too large to be processed on a single disk
- Cloud computing has emerged as the key technology to enable large-scale analytics
- Many open challenges
  - Graph data is not typically amenable to distributed processing
  - How to represent complex analysis tasks declaratively ?
- We will cover this late, but a good topic for class projects

Stuff

- Format of the class
  - Assigned paper readings
  - Critiques must be posted on the forum before the class
- Class Projects
  - Major component of the grade
  - Individual or Group
  - Intermediate “literature survey” deadline in mid-October
  - I will provide suggestions; your own projects welcome

Stuff

- Format of the class
  - Assigned paper readings
  - Critiques must be posted on the forum before the class
- Class Projects
  - Major component of the grade
  - Individual or Group
  - Intermediate “literature survey” deadline in mid-October
  - I will provide suggestions; your own projects welcome
- My absences
  - Gone a little over 2 weeks in September
  - Those classes would be covered by students
  - We also miss four other classes (labor day, thanksgiving, two colloquium talks)
  - Instead of rescheduling, prefer to go longer in other classes (by 15 to 30 mins)

Next class. . .

- Probabilistic Databases
  - Reading: On the webpage