IR & DB:
Toward Controversy and Philosophy

Joe Hellerstein
UC Berkeley & Intel Research
Controversial Statements
Controversial Statements

• Surajit was right
Controversial Statements

• Surajit was right
• I am an expert on this stuff
Controversial Statements

• Surajit was right
• I am an expert on this stuff
  • Led integration of AltaVista and Cohera federated DB
  • Keyword search, Q-grams, auto-taxonomization, visual data cleaning & integration, etc.
    • With consulting from Martí Hearst (IR meets DB!)
• [H & S, SIGMOD ’01 industrial track]
Controversial Statements II

• This is mostly busyness, some re-search
Controversial Statements II

• This is mostly busyness, some re-search
  • Identify BKMs
  • Perform SI
• Microsoft will work on TCO
Controversial Statements II

- This is mostly busyness, some re-search
  - Identify BKMs
  - Perform SI
  - Micro$oft will work on TCO
- N$F?

- SO: Let’s “Uplevel the Dialogue”
Let’s Get Philosophical

• We have a narrow definition of structure
• Structure is not about Relational Databases vs. Text Databases vs. “semi-structured”
• In fact, even text is very structured!
Let’s Get Philosophical

• We have a narrow definition of structure
• Structure is not about Relational Databases vs. Text Databases vs. “semi-structured”
  • In fact, even text is very structured!

• Human discourse based on “deep structure”
  • Ferdinand De Saussure, the father of Structural Linguistics
  • Extended by Chomsky
  • Also speak of “relational” nature of language
So what’s the philosophical diff?

• Twofold
  • The SOURCE of the structure
  • The USE of the structure
Relational Databases

- **ENGINEERED STRUCTURE**
  - The design of structure is at the heart of the discussion
  - Codd’s lessons: simple design for robust evolution

- **LANGUAGE AS PROGRAMMATIC INTERFACE**
  - Semantically strict queries provide predictable results
  - Suitable for computer interpretation
    - Programmer can reason about invariants
  - Hence good for embedding into application code
    - Relatively few users interface directly to a Database!
Information Retrieval

• “FOUND” STRUCTURE
  • Take a pile of information, and extract structure therefrom

• HUMAN-CENTRIC EXPLOITATION
  • Rough understanding of query intent
  • Interface more important than query language
    • User can browse/filter/interpret some results
  • Requires a human in the loop
    • Relatively few programs embed IR techniques invisibly
The Synergy

• This is not about semi-structured data!!!
It’s not that semi-structured is bad...
It’s not that semi-structured is bad...

It’s just that semi-structured is not semi-structured
It’s not that semi-structured is bad...

It’s just that semi-structured is not semi-structured

_Ceci n’est pas une pipe._
It’s not that semi-structured is bad...

It’s just that semi-structured is not semi-structured!
The Synergy

- **DB folk working on FOUND STRUCTURE**
  - E.g. text queries on “structured” data
  - Information extraction
  - User-centric data mining
  - Schema corpora

- **DB folk working on HUMAN INTERFACES for handling uncertainty**
  - E.g. keyword search of structure data
  - E.g. online aggregation, visual data cleaning, cube navigation, other user-centric mining tasks
The Synergy

- **IR folk working on ENGINEERED STRUCTURE**
  - E.g. document design
  - But the simpler, the better (Codd)!!

- **IR folk working on embedded systems**
  - Google SOAP interface clients (Googlisms)
  - ??

- Frankly, I’m out of my comfort zone
For More Rambling on this Topic

• See the WebDB keynote on my home page
  • http://www.cs.berkeley.edu/~jmh
The Inktomi Search Engine

Result Set = [DocId, Score, URL, Date, Size, Abstract]

Top(k, Score) ⊙ DocId

[DocId, Score] ⊙ [DocId, URL, Date, Size, Abstract]

c1 Quality(D)

docId

score = Quality(D)

matching documents:

score = \sum_{i} W_i Score(w_i, d)

score = +c_2 \sum_{i} W_i Score(w_i, d)