Advancing Collaboration – Efforts by Database Community

Presented By
Gautam S. Thakur
Main Idea

• Collecting Information is relevant to a group

• Using specialized database tools and techniques

• Processing raw data and presenting in a structure form to run queries

• Sharing and specializing.
Agenda

• DBLife: A Community Information Management Platform for the Database Research Community

• Computational Journalism – A Call to Arms to Database Researchers
DBLife: A Community Information Management Platform for the Database Research Community
Authors

- Pedro DeRose
- Warren Shen
- Fei Chen
- Yoonkyong Lee
- Doug Burdick
- AnHai Doan
- Raghu Ramakrishnan
Goals and Challenges

• Community members often want to
  – Discover, monitor, and query entities and relationships in their community.

• Involves
  – retrieving raw, largely unstructured data from multiple sources (e.g., home pages, DBLP, mailing lists),
  – inferring and monitoring semantic information.
Problem – Community Information Management

• Web is simply too big.

• Difficult to build infrastructures and user bases at the Web scale to uncover more interesting problems and better validate solutions.

• Examples –
  – “J. Gray” and “Jim Gray” are same
  – Recognizing a relationship (e.g., co-authoring, advising)
  – Inferring that a relationship has ceased to exist.
Solution

• Cimple Project - joint project between the University of Wisconsin and Yahoo! Research
• Develop a software platform
• Data-rich online community can quickly deploy and customize to effectively manage its data.
• Domains
  – AILife, IRLife etc.
Cimple’s Three Step Approach

• Start with High Quality Seed
  – Relevant data sources and domain knowledge
  – Entities and relationships of interest

• Automatic methods to create and maintain an entity-relationship graph of the community.

• Providing valuable, carefully crafted functionalities
  – helps correct, maintain, and evolve relationships and this ER graph.
Generally Speaking

• Apply database technologies to manage Web data.

• Circumvent problem by focusing on Web communities, which are in effect “mini-Webs”.
The DBLife System

- dblife.cs.wisc.edu
- 1.5 years
- Nearly 900 data sources, downloads 9,500 pages, or 150+ MB, daily
- Tracks roughly 335,000 mentions of 16,600 entities
- DBLife at a SIGMOD-06, CIDR-07.
Rest of the talk

- DBLife’s features
- DBLife’s internals
- Range of open research issues in CIM
DBLife Workflow

- High-Quality Seed
- Automatic Methods
- Leveraging the Community

Diagram showing layers and connections within a database system.
DBLife Features

• Newsletter – *currently discontinued*
• Superhomepages
• Mass Collaboration
• Event Trackers
• New Features
DBLife Internals - Architecture

• Architecture
  – extensible set of independent but intercommunicating modules.

• XML based data handling, common to all modules

• Central configuration file specifies the modules’ input and order of execution;
  – when the system is run, the configuration file is interpreted,
  – modules are executed in turn.

• After each execution, output is archived. (later used)
DBLife Internals - Architecture

• **Core Modules**
  – Responsible for creating and maintaining the system’s ER graph

• **Application Modules**
  – Exploits the ER graph to generate output displayed by DBLife’s web interface
DBLife Internal - Workflow

- High-Quality Seed
  - Information from Community Experts
  - Initial list of sources to crawl
  - Domain knowledge
DBLife Internal - Workflow

- Automatic Methods - to create and maintain the ER graph
  - Rely on “heavily” domain knowledge
  - As of now algorithms, future $\rightarrow$ declarative languages

- Core Modules are organized in Three (3) Layers
  - *Data page*
  - *Mention*
  - *Entity*
DBLife Internal - Workflow

• Layer 1 – *data pages*
  – Crawl the specified sources, cache downloaded pages
  – Update old cache

• Modules for
  – extracts metadata
  – detect structural elements within pages
  – Used to specify domain knowledge for the purpose of filtering.
DBLife Internal - Workflow

• Layer 2 - *Entity Mention*
  – Detect dictionary of names to find mentions
  – Map old mentions, detect new ones
  – *Mention Tracking*

• Vital for ER graphs – Detect change

• Modules for
  – Finding metadata for each mention → appearing
Layer 3 – *Entity layer*

- Matches mentions to one another, disambiguates them, and groups them into entities.
- Infers relationships between entities using domain knowledge rules
- Creating an entity-relationship graph
- Current set of entities is reconciled with those from the previous execution (Very difficult)
  - J Gray and Jim Gray
  - Change research affiliation
Automatic Methods - to create and maintain the ER graph
  - Rely on “heavily” domain knowledge
  - As of now algorithms, future → declarative languages

Core Modules are organized in Three (3) Layers
  - Data page
  - Mention
  - Entity
DBLife Internal - Workflow

• Leveraging the Community
  – Automatic methods are inherently imperfect
  – Leverages the community through mass collaboration
    • Correct errors and maintain ER graph
  – Incentive to update
    • Vote on the correct pictures
    • Edit super home pages
Open Research Issues

• Efficient information extraction
• Integrate information
• Construct provenance, explanation, and uncertainty mechanisms that work for such contexts
• Perform better disambiguation of inferred data; and
• Maintain the extracted data as the underlying raw data evolves
Demo of DBLife
Computational Journalism – A Call to Arms to Database Researchers
Authors

• Sarah Cohen
• Chengkai Li
• Jun Yang
• Cong Yu

• A workshop at Stanford on the nascent field of computational journalism
• CIDR, 2011
Challenges

• Change in the journalism efforts
  – Cost intensive, but declining readerships – News paper
  – Way information is digested is changing
  – Gathering reports than generating by reporters

• Transformed into
  – Aggregators

• Danger - original, in-depth investigative reporting will fade
Solution

• Investigative & Computation Journalism
  – Lower cost, increase effectiveness, encourage participation,

• In this work
  – A system to support mass collaboration
  – Bringing reports and citizens.
  – From DB community perspective
A cloud for the crowd

- Combines computational resources as well as human expertise
- Support more efficient and effective investigative journalism.
- Rent infrastructure, map-reduce for scalable data processing
  - Example, NY time images to pdfs.
Vision

• Example – DocumentCloud.org - NYT

• *Manage, integrate, and analyze* structured *data*

• From Structure data draw
  – Declarative languages, continuous querying, to parallel query processing.
Crowd sourcing

• Never be fully automated

• Leverage Human expertise
  – Annotations,
  – Guardian example,

• *Goal driven Crowd* sourcing
  – Converting the big task to mini tasks
Mini Task

• Checking an entity-relationship extracted a document

• Reconciling entries from two different public databases

• Listing and Crow Sourcing - mini task based on user expertise and preferences

• Completion Importance → High Priority
Challenges Intelligent Crowd Sourcing

- Ways to represent *Prior knowledge, uncertainty in the data* → reasoning with it.
  - Using **Probabilistic** database – small sources !!

- **Sensitivity** Analysis – which data matter most?

- **Lineage** – to track original sources

- **Value of perfect information**
Cost Quantification

• Acquisitional query processing

• dynamically based on feedback

• Leverage and augment human expertise
Recap

- Cloud for Crowd
- Hosting tools and data
- Runs computation tasks
- Intelligently plan and manage crowd sourcing
- Augmenting computer + Human Expertise

- Next, let's talk about specific Database Challenges
From finding answers to finding questions

• ... Hopkins County also maintained the lowest monthly jobless rate in the immediate eight-county region for the 29th consecutive month. ... [11].

– Pointer to the relevant data source
– Query

• Turning stories live
  – By continuously evaluate the query as the data source is updated, and alert us when its answer changes.
Features

- Alert systems
  - For annual reports, temperatures etc.

- Stories multiply
  - Parameterized SQL Query template
  - Lead to other stories

- Fact check
  - Exposes mis-information
  - Guards against errors
  - Fact checking – through crowd sourcing.
Underlying assumptions

• Sources and queries to support claims

• How to get it?
  – Makers to provide it – may not work

• System help us identify the data source and reverse-engineer the query associated with a claim?
  – Through NLP
Two New approaches

• Text not only for the query, but also for its answer.

• More people $\rightarrow$ more usage $\rightarrow$ system build over the time, reverse engineering queries to know relevant and genuine sources
Reporter’s black box

- Hand crafted queries on a structure database
- Queries of form
  - Grouping, aggregation, ranking, looking for outliers, checking for missing values, etc.
- Automatically monitor
- Find patterns across database
- Newsworthiness – sorting queries that are relevant
  - Here comes the cloud and parallelization to run queries.
Perspective

• Creating a ‘mini-web’ for specialized purposes.

• A support collaborative investigative journalism.

• Privacy, trust and authority, data mining, information retrieval, speech and vision, visualization, etc.
Thanks

• Questions.