Data-Centric
Human Computation

Jennifer Widom
Stanford University
Mega-Trends in C.S. Research

2000’s
Crowd
2010’s
Human Computation

Augmenting computation with the use of human abilities to solve (sub)problems that are difficult for computers

- Object/image comparisons
- Information extraction
- Data gathering
- Relevance judgments
- Many more...

≈ “Crowdsourcing”
Crowdsourcing Research

- Complex ops: Sort, Cluster, Clean
- Basic ops: Compare, Filter

Data Gathering / Query Answering

Algorithms

Platforms

- Get data
- Verify
- Interfaces
- Incentives
- Trust, reputation
- Spam
- Pricing

Marketplace #1

Marketplace #2

... Marketplace #n

Jennifer Widom
New Considerations & Tradeoffs

- How long can I wait?
- How much am I willing to spend?
- What is my desired quality?

Latency

Uncertainty
Live Experiment – Human Filter

Are there more than 40 dots?
Computing the Answer

• **Yes or No?**

• **With what confidence?**

• **Should I ask more questions?**
  - More cost (−)
  - Higher latency (−)
  - Higher accuracy (+)
Live Experiment – Filter #2

Are more than half of the dots blue?
Live Experiment – Two Filters

Are there more than 40 dots and are more than half of the dots blue?
Computing the Answer

• Ask questions separately or together?
  Together ⇒ lower cost, lower latency, lower accuracy?

• If separately, in sequence or in parallel?

• If in sequence, in what order?
  Different filters may have...
  ▪ different cost
  ▪ different latency
  ▪ different accuracy
Crowd Algorithms

Design fundamental algorithms involving human computations

- Filter a large set (human predicate)
- Sort or find top-k from a large set (human comparison)

- Which questions do I ask of humans?
- Do I ask sequentially or in parallel?
- How much redundancy in questions?
- How do I combine answers?
- When do I stop?
Algorithms We’ve Looked At

- **Graph search** [VLDB ‘11]
- **Filtering** [SIGMOD ‘12]
- **Find-Max** [SIGMOD ‘12, WWW ‘12]
- **Entity Resolution** (= Deduplication)
Sample Results: Filtering

**Given:**
- Large set $S$ of items
- Filter $E$ over items of $S$
- Selectivity $\sigma$ of $E$ on $S$
- Human false-positive rate $\rho$
- Human false-negative rate $\nu$

**Find:** a strategy for asking $E$ that:
- Asks no more than $m$ questions per item
- Guarantees overall expected error less than $e$
- Minimizes overall expected cost $c$ (# questions)

**Strategies:**
- Exhaustive search
- Pruned search
- Probabilistic strategies
Crowdsourcing Research

Data Gathering / Query Answering

Algorithms

Platforms

Marketplace #1

Marketplace #2

... ... ...

Marketplace #n
Find the capitals of five Spanish-speaking countries

<table>
<thead>
<tr>
<th>Country</th>
<th>Capital</th>
<th>Language</th>
</tr>
</thead>
<tbody>
<tr>
<td>Peru</td>
<td>Lima</td>
<td>Spanish</td>
</tr>
<tr>
<td>Peru</td>
<td>Lima</td>
<td>Quechua</td>
</tr>
<tr>
<td>Brazil</td>
<td>Brasilia</td>
<td>Portugese</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

- Give me a Spanish-speaking country
- What is the capital of country X?
- What language do they speak in country X?
- Give me a country
- Give me a capital
- Is this country-capital-language triple correct?
- Give me the capitals of five Spanish-speaking countries
Human-Powered Query Answering

Find the capitals of five Spanish-speaking countries

<table>
<thead>
<tr>
<th>Country</th>
<th>Capital</th>
<th>Language</th>
</tr>
</thead>
<tbody>
<tr>
<td>Peru</td>
<td>Lima</td>
<td>Spanish</td>
</tr>
<tr>
<td>Peru</td>
<td>Lima</td>
<td>Quechua</td>
</tr>
<tr>
<td>Brazil</td>
<td>Brasilia</td>
<td>Portugese</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

Inconsistencies

- What if some humans say Brazil is Spanish-speaking and others say Portugese?
- What if some humans answer “Chile” and others “Chili”?
Key Elements of Our Approach

- Exploit relational model and SQL query language
- Configurable fetch rules for obtaining data from humans
- Configurable resolution rules for resolving inconsistencies in fetched data
- Traditional approach to query optimization
  - But with many new twists and challenges!
Deco: Declarative Crowdsourcing

1) Schema for conceptual relations
   restaurant(name, cuisine, rating)

2) Fetch rules
   name ⇒ cuisine: f1(n)
   cuisine, rating ⇒ name: f2(c, r)

3) Resolution rules
   name: dedup(N)
   rating: average(R)
Deco: Declarative Crowdsourcing

Declarative queries
Select name From restaurant
Where cuisine = ‘Thai’
And rating >= 3
Atleast 5

Query semantics
Relational result over
“some valid instance”

Valid instance
fetch + resolve + join

User or Application
DBMS
Deco: Declarative Crowdsourcing

Generate query execution plan that orchestrates and optimizes fetches and resolutions to produce answer

Different possible objectives:
- $N$ tuples, minimize cost (fetches)
- $F$ fetches, maximize tuples
- $T$ time, minimize/maximize ??

Latency

Uncertainty

Cost
Deco: Declarative Crowdsourcing

Query Processor

DBMS

Deco: A System for Declarative Crowdsourcing

SELECT * FROM rest WHERE cuisine='French' ATLEAST 10

Depto: Stanford University InfoLab © 2012

Jennifer Widom
Crowdsourcing Research

Humans As Data Processors

Data Gathering / Query Answering

Algorithms

Platforms

Marketplace #1

Marketplace #2

Marketplace #n

Humans As Data Providers

Jennifer Widom
Data-Centric Human Computation

Joint work with:
Hector Garcia-Molina, Neoklis Polyzotis, Aditya Parameswaran, Hyunjung Park