Collaborative Querying & Visualizing to improve on standard Information Retrieval

David Crowe
Improving Information Retrieval (IR)

• Problem:
  o missing/unhelpful results

• Suppose a user searches for "Subway"…
Improving Information Retrieval (IR)

• Problem:
  o missing/unhelpful results

• Causes:
  o users unfamiliar with Information Retrieval (IR) ops
  o vocab mismatches
  o context/semantics

• To Solve it:
  o Use past queries to identify strongly-related queries
  o Show these similar queries, let user explore/learn
Categorizing Similarity

• **Term-Based:** (query = "bag of terms")
  - query attributes ONLY (*Query Term Vectors*)

• **Result-Based:** (query = "result of executing")
  - result attributes ONLY (*Term Vectors, URLs*)

• **Feedback-Based:** (query = "relevant results")
  - result attributes AND clickthrough-data

• **Community-Based:** (consider interests of the user)
  - subsets of queries by user affiliation
Google's Search & Ad Data

Your categories

Below you can review the interests and inferred demographics that Google has associated with your cookie. You can remove or edit these at any time.

Computers & Electronics - Software - Internet Software - Internet Clients & Browsers
Computers & Electronics - Software - Operating Systems
Computers & Electronics - Software - Operating Systems - Linux & Unix
Computers & Electronics - Software - Operating Systems - Mac OS
Games - Computer & Video Games
Pets & Animals - Pets - Cats

Your demographics

We infer your age and gender based on the websites you've visited. You can remove or edit these at any time.

Age: 25-34
Gender: Male

Supposed to be at:
https://www.google.com/settings/ads/onweb/
How Query Graph Visualization does similarity + clustering

- To avoid term/result-based drawbacks, QGV defines 'hybrid_similarity' for queries Qi,Qj (ALPHA+BETA=1):
  - hybrid_similarity(Qi,Qj) = ALPHA*result_similarity(Qi,Qj) + BETA*term_similarity(Qi,Qj)

- A cluster on some node Qi is a list of nodes Qj that are similar by more than 'some number' THRESHOLD:
  - hybrid_similarity(Qi,Qj) ≥ THRESHOLD

- The team found the best result given when:
  - ALPHA = 0.75 | BETA = 0.25 | THRESHOLD = 0.9
The Query Graph (QGV)

Functionality:
- Generate clusters to form a Query Network graph
- Allow users to explore the graph visually

Visualization:
- cluster = directly connected
- root query = white
- depth from root = lightness
- similarity = edge coefficient
Navigating and Searching

- **Toolbar:**
  - zoom: shrink/grow view
  - rotate: view from different directions
  - locality zooming: set network depth to draw

- **Node Controls (popup):**
  - Search via outside IR provider
  - Make Root Node
  - Expand/Collapse
QGV Displaying Clusters (Visual)
<table>
<thead>
<tr>
<th>Heuristic</th>
<th>Average score&lt;sup&gt;a&lt;/sup&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>Visibility of system status</td>
<td>4.0</td>
</tr>
<tr>
<td>Match between system and real world</td>
<td>4.1</td>
</tr>
<tr>
<td>User control and freedom</td>
<td>3.6</td>
</tr>
<tr>
<td>Consistency and standards</td>
<td>4.2</td>
</tr>
<tr>
<td>Error prevention</td>
<td>4.2</td>
</tr>
<tr>
<td>Recognition rather than recall</td>
<td>4.1</td>
</tr>
<tr>
<td>Flexibility and efficiency of use</td>
<td><strong>4.5</strong></td>
</tr>
<tr>
<td>Aesthetic and minimalist design</td>
<td>4.2</td>
</tr>
<tr>
<td>Help user recognize, diagnose and recover from errors</td>
<td>3.9</td>
</tr>
<tr>
<td>Help and documentation</td>
<td><strong>2.3</strong></td>
</tr>
</tbody>
</table>

**Table I.**
Summary of heuristic evaluation results

Note: <sup>a</sup>1 = strongly disagree, 5 = strongly agree
Do you think this could be extended to SQL?
Any application to your projects?
Can we take it further?
Did they achieve their goal?
Other thoughts

• Link to paper 'Nielsen' ratings (Evaluation):
  http://dl.acm.org/citation.cfm?doid=191666.191729
  I haven't read it, but it sounded interesting.

• Optimization:
  o Given a query Q in the DB that matches some result document, replace the document with Q [document surrogate] since it is a fair description. In tests this boosts performance by almost 30%. (Billerbeck et al. (2003))