Refactoring the Search Problem

Gary William Flake
Microsoft Corporation
But a better title may have been...

Zoomable UIs, Information Retrieval, and the Uncanny Valley
Outline

• DeepZoom and Pivot Demos
• Web search retrospective
• ZUIs and similarities to IR
• Refactoring Web search
• Pivot + search demo
• Closing remarks
What’s worked well in search

• Instant answers
• Spell correction
• Vertical tabs
• Query suggestions
• Query completion
• Facets (vertical)
• Grouping results
• Paid search
What’s worked well in search

• Instant answers
• Spell correction
• Vertical tabs
• Query suggestions
• Query completion
• Facets (vertical)
• Grouping results
• Paid search

• Biggest improvement is in overall index scale
• Some improvement in core relevance
What’s worked well in search

• Instant answers
• Spell correction
• Vertical tabs
• Query suggestions
• Query completion
• Facets (vertical)
• Grouping results
• Paid search

• Biggest improvement is in overall index scale
• Some improvement in core relevance

But this list is actually pretty modest
What hasn’t worked as well

- Natural language queries
- Richer presentations for results
- Richer presentation for one result
- Clustering (visual or otherwise)
- Relevance feedback
- Social search over organic web
- The next page button
Grokker (RIP, 2009)
Typical web search formulation

1. Information Need
2. Query
3. Send to System
4. Receive Results
5. Evaluate Results
6. Reformulate
7. Done?
8. Stop
Atypical web search formulation

1. Information Need
2. Query
3. Reformulate
4. Send to System
5. Receive Results
6. Evaluate Results
7. Done?
8. Stop

Fluid Interaction
The uncanny valley

- familiar

human likeness

- humanoid robot
- stuffed animal
- corpse
- prosthetic hand
- healthy person
- bunraku puppet

moving

still
## Discrete vs. continuous interactions

<table>
<thead>
<tr>
<th>Interaction Bandwidth</th>
<th>CGI</th>
<th>Games</th>
<th>Communication</th>
<th>User interfaces</th>
<th>Search</th>
</tr>
</thead>
<tbody>
<tr>
<td>Low</td>
<td>Stick figures</td>
<td>Text games</td>
<td>Post cards</td>
<td>Text terminal</td>
<td>Grep</td>
</tr>
<tr>
<td>High</td>
<td>The Simpsons</td>
<td>Asteroids</td>
<td>Email</td>
<td>Web 1.0</td>
<td>AltaVista</td>
</tr>
<tr>
<td></td>
<td>Toy Story</td>
<td>Halo</td>
<td>Phone</td>
<td>Rich client</td>
<td>Present day search engines</td>
</tr>
<tr>
<td></td>
<td>Polar Express</td>
<td>Final Fantasy</td>
<td>Video conferencing</td>
<td>Over ambitious AJAX</td>
<td>Grokker</td>
</tr>
<tr>
<td></td>
<td>Avatar</td>
<td>In-person sports</td>
<td>Face-to-face</td>
<td>Good ZUIs</td>
<td>???</td>
</tr>
</tbody>
</table>

**Communication:**
- Post cards
- Email
- Phone
- Video conferencing
- Face-to-face

**User interfaces:**
- Text terminal
- Web 1.0
- Rich client
- Over ambitious AJAX
- Good ZUIs

**Search:**
- Grep
- AltaVista
- Present day search engines
- Grokker
- ???
Surpassing the uncanny valley is exceedingly difficult because it requires excellence in all of science, technology, and design.
Our dilemmas

We are already familiar with the dilemma of **precision versus recall**.

There exists a similar dilemma around **scale**, **fluidity**, and **complexity**.
Outline

• DeepZoom and Pivot Demos
• Web search retrospective
• **ZUIs and similarities to IR**
• Refactoring Web search
• Pivot + search demo
• Closing remarks
DeepZoom items (DZI)

- Each tile is an image file
- Each level is a set of image files in a folder
- Each pyramid is a set of folders with image tiles for each level
- A simple XML file describes the size of the image
- Images can have only some tiles (sparse images)
- Pyramid 1.4x original size
- 6Mpixels converts in 1.5s
DeepZoom collections (DZC)

- Thumbnails are packed into shared tiles
- Loading 100’s of images requires loading few tiles
- Patented Morton layout makes adding items $O(1)$
- No more than one tile per level is not full
- Optimizes rendering by sharing textures
- Storage is roughly one thumbnail per image
End-to-end details

• Back-end preprocesses the assets
  • Once for static items
  • In real-time for dynamic items
• Front-end stitches the pieces
• UI is ideally asynchronous in every way
• Only recently possible (bandwidth and GPUs)
• Net outcome is that user feels in control
Why user control is essential

• They feel empowered to explore
• Actions are more clearly invertible
• Mistakes can be overcome
• More robust mental models are possible
## Comparison: ZUIs and IR

<table>
<thead>
<tr>
<th></th>
<th>ZUIs</th>
<th>IR</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>End Goal</strong></td>
<td>Find a point of interest</td>
<td>Find an item of interest</td>
</tr>
<tr>
<td><strong>Challenge</strong></td>
<td>Scale and speed</td>
<td>Precision and recall</td>
</tr>
<tr>
<td><strong>Specificity</strong></td>
<td>Zoom in</td>
<td>Narrow or filter</td>
</tr>
<tr>
<td><strong>Generality</strong></td>
<td>Zoom out</td>
<td>Broaden or un-filter</td>
</tr>
<tr>
<td><strong>Similarity</strong></td>
<td>Pan about</td>
<td>Related items</td>
</tr>
<tr>
<td><strong>Points of Interest</strong></td>
<td>Landmarks</td>
<td>Suggested filters</td>
</tr>
<tr>
<td><strong>Uncanny Valley</strong></td>
<td>Low frame rate</td>
<td>Breaking expectations</td>
</tr>
<tr>
<td><strong>Ideal</strong></td>
<td>User engagement</td>
<td>User engagement</td>
</tr>
</tbody>
</table>
Lessons from ZUIs to apply to IR

• Preprocess on the back-end.
• Assume that the font-end can do a lot.
• Build UI around continuous interactions.
• Use asynchronous I/O between endpoints.
• Use the two in combination to reinforce one another (left versus right brain).
Outline

- DeepZoom and Pivot Demos
- Web search retrospective
- ZUIs and similarities to IR
- Refactoring Web search
- Pivot + search demo
- Closing remarks
Higher level goals

• Turn the present discrete mode of interaction of search into a continuous dialogue.
• Supports fluid interactions that are powerful, informative, and fun.
• Scale to thousands of items within the user / client interactions.
• Be the second best UI for any collection.
Web Search Architecture

- **User and Browser Client**
- **Web Frontend**
- **Web Backend**

**Components:**
- Streaming Data Payload
- Crawler
- Indexer
- Cache
- Query Processor
- Search Front End
- Index
- Static Ranking & Analysis
- Dynamic Ranking & Analysis
- Fluid Interaction

**Processes:**
- Query & Other acts
- Result set
Pivot UI features
Pivot UI features

Facet
Groups
Pivot UI features

Facets
Pivot UI features

Info Pane
(Supplement)
Data payload sent to client

Facet Groups  Facets  Items  Supplements

This part of the bipartite graph is where most of the cost is found.
Technical details

- Most critical part to the UI is very basic meta data for **facets**, **facet groups**, and **items**.
- Edges from facets to items can be streamed asynchronously with the UI updated dynamically.
- **Supplements** are only needed when item is zoomed in.
- Biggest challenge is in dynamic generation of collections on the backend.
Server-side IR problems

- Basic ranking
- Facet determination
- Cleaning bipartite graph
- Augmenting bipartite graph
- Broadening facets for info pane only
- Cross-collection references (for collections of collections)
Client-side IR problems

- Clustering
- Filter suggestions
- Filter and broaden
- Query suggestions
- Re-ranking via filters
- Item-to-item similarity
- Group-to-group similarity
- Collection-wide visualization
Outline

• DeepZoom and Pivot Demos
• Web search retrospective
• ZUIs and similarities to IR
• Refactoring Web search
• **Pivot + search demo**
• Closing remarks
First: do no harm

- Linear order must be obvious
- First result or instant answer is prominent
- First 4 or so items are easily visible
- Preserve title / description / URL format
- Elegant iteration over many items
- Don’t do something so different that it offends the user
Next: modestly improve

- Handle many more results (> 50)
- Basic n-grams extraction
- Filter / broaden over snippet text
- Filter / broaden over domains
- Allow operations over entire session
- Bonus: view can be repurposed
Not done

• Document classes as facets
• Document similarity as synthetic facets
• Folksonomies and community tags
• Federation over verticals
• Many more sophisticated algorithms
Outline

• DeepZoom and Pivot Demos
• Web search retrospective
• ZUIs and similarities to IR
• Refactoring Web search
• Pivot + search demo

• Closing remarks
The virtuous cycle of the Web

Consumption
  Enhanced Modes of Discovery

Connections
  New Value in Combining Data

Creations
  Simplified Authoring & Participation
The vicious cycle of the Web

• Easier to create → more people create
• More stuff created → harder to find good stuff
• Quality not as rewarded → lower standards
• Lower quality content → shorter attention spans
• Short attention span → reduce standards further
What’s the cure?

We desperately need a mode of interaction where the whole of the data is greater than the sum of the parts.

Wisdom > knowledge > information > data
Playing with Pivot

Client Application (Experimental)

• Software installation.
• Requires Vista or Windows 7.
• Basic authoring tools available now.

Silverlight Embed (Production)

• Runs within web page.
• Works anywhere
• Partners spinning up advance authoring.

Both available at http://getpivot.com/
Thanks!