How far do people look for results?

(Source: iprospect.com WhitePaper_2006_SearchEngineUserBehavior.pdf)

“When you perform a search on a search engine and don’t find what you are looking for, at what point do you typically either revise your search, or move on to another search engine? (Select one)"

- 27% - After reviewing the first few entries
- 25% - After reviewing the first page
- 16% - After reviewing the first 2 pages
- 12% - After reviewing the first 3 pages
- 20% - After reviewing more than 3 pages
Users’ empirical evaluation of results

- Quality of pages varies widely
  - Relevance is not enough
  - Other desirable qualities (non IR!!)
    - Content: Trustworthy, diverse, non-duplicated, well maintained
    - Web readability: display correctly & fast
    - No annoyances: pop-ups, etc

- Precision vs. recall
  - On the web, recall seldom matters

- What matters
  - Precision at 1? Precision above the fold?
  - Comprehensiveness – must be able to deal with obscure queries
    - Recall matters when the number of matches is very small

- User perceptions may be unscientific, but are significant over a large aggregate

---

Users’ empirical evaluation of engines

- Relevance and validity of results
- UI – Simple, no clutter, error tolerant
- Trust – Results are objective
- Coverage of topics for polysemic queries
- Pre/Post process tools provided
  - Mitigate user errors (auto spell check, search assist,...)
  - Explicit: Search within results, more like this, refine ...
  - Anticipative: related searches

- Deal with idiosyncrasies
  - Web specific vocabulary
    - Impact on stemming, spell-check, etc
  - Web addresses typed in the search box
Search results refinement methods

- **Relevance feedback**
  - Rocchio algorithm
  - Blind-relevance feedback

- **Query expansion**
  - Linguistic expansion
  - Statistical expansion

---

**Relevance feedback**

- Given the initial search results, the user marks some images as important or non-important.
  - This information is used for a second search iteration where these examples are used to refine the results

- The characteristics of the positive examples are used to boost documents with similar characteristics
- The characteristics of the negative examples are used to penalize document with similar characteristics

- Also known as “active learning”, “online learning”, “reinforcement learning” and “semi-automatic search”
### Results for Initial Query

<table>
<thead>
<tr>
<th>Image</th>
<th>Relevance Score</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1.jpg" alt="Image 1" /></td>
<td>0.0</td>
</tr>
<tr>
<td><img src="image2.jpg" alt="Image 2" /></td>
<td>0.0</td>
</tr>
<tr>
<td><img src="image3.jpg" alt="Image 3" /></td>
<td>0.0</td>
</tr>
<tr>
<td><img src="image4.jpg" alt="Image 4" /></td>
<td>0.0</td>
</tr>
<tr>
<td><img src="image5.jpg" alt="Image 5" /></td>
<td>0.0</td>
</tr>
<tr>
<td><img src="image6.jpg" alt="Image 6" /></td>
<td>0.0</td>
</tr>
<tr>
<td><img src="image7.jpg" alt="Image 7" /></td>
<td>0.0</td>
</tr>
<tr>
<td><img src="image8.jpg" alt="Image 8" /></td>
<td>0.0</td>
</tr>
</tbody>
</table>

### Relevance Feedback

<table>
<thead>
<tr>
<th>Image</th>
<th>Relevance Score</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1.jpg" alt="Image 1" /></td>
<td>0.0</td>
</tr>
<tr>
<td><img src="image2.jpg" alt="Image 2" /></td>
<td>0.0</td>
</tr>
<tr>
<td><img src="image3.jpg" alt="Image 3" /></td>
<td>0.0</td>
</tr>
<tr>
<td><img src="image4.jpg" alt="Image 4" /></td>
<td>0.0</td>
</tr>
<tr>
<td><img src="image5.jpg" alt="Image 5" /></td>
<td>1.0</td>
</tr>
<tr>
<td><img src="image6.jpg" alt="Image 6" /></td>
<td>0.0</td>
</tr>
<tr>
<td><img src="image7.jpg" alt="Image 7" /></td>
<td>0.0</td>
</tr>
<tr>
<td><img src="image8.jpg" alt="Image 8" /></td>
<td>0.0</td>
</tr>
</tbody>
</table>

Note: The relevance scores are based on the relevance feedback provided by the user.
Results after Relevance Feedback

Ad hoc results for query canine

ad hoc retrieval canine
cat dog

"canine"
Ad hoc results for query canine

User feedback: Select what is relevant
Results after relevance feedback

Key concept: Centroid

- The centroid is the center of mass of a set of points
- Recall that we represent documents as points in a high-dimensional space
- Definition: Centroid

$$\bar{\mu}(C) = \frac{1}{|C|} \sum_{d \in C} \bar{d}$$

where $C$ is a set of documents.
Rocchio Algorithm

- The Rocchio algorithm uses the vector space model to pick a relevance feedback query.
- Rocchio seeks the query $\mathbf{q}_{opt}$ that maximizes

$\tilde{q}_{opt} = \arg \max_{\mathbf{q}} [\cos(\tilde{q}, \mu(C)) - \cos(\tilde{q}, \mu(C_w))]$

- Tries to separate docs marked relevant and non-relevant

$\tilde{q}_{opt} = \frac{1}{|C_r|} \sum_{d \in C_r} \tilde{d}_j - \frac{1}{|C_{nr}|} \sum_{d \notin C_r} \tilde{d}_j$

- Problem: we don’t know the truly relevant docs

---

The Theoretically Best Query

The diagram shows the concept of the theoretically best query in Rocchio’s algorithm. The optimal query is depicted as a line that attempts to separate relevant documents (circles) from non-relevant documents (Xs). The diagram illustrates how the algorithm seeks to maximize the relevance feedback query by adjusting the query based on the feedback received.
Relevance feedback on initial query

Initial query

Revised query

x known non-relevant documents
○ known relevant documents

Sec. 9.1.1 Rocchio 1971 Algorithm (SMART)

- Used in practice:
  \[
  \tilde{q}_m = \alpha \tilde{q}_0 + \beta \frac{1}{|D_r|} \sum_{d_j \in D_r} \tilde{d}_j - \gamma \frac{1}{|D_{nr}|} \sum_{d_j \in D_{nr}} \tilde{d}_j
  \]

- \(D_r\) = set of known relevant doc vectors
- \(D_{nr}\) = set of known irrelevant doc vectors
  - Different from \(C_r\) and \(C_{nr}\)
- \(q_m\) = modified query vector; \(q_0\) = original query vector;
  \(\alpha, \beta, \gamma\): weights (hand-chosen or set empirically)
- New query moves toward relevant documents and away from irrelevant documents
Subtleties to note

- Tradeoff $\alpha$ vs. $\beta/\gamma$: If we have a lot of judged documents, we want a higher $\beta/\gamma$.

- Some weights in query vector can go negative
  - Negative term weights are ignored (set to 0)

Relevance Feedback: Assumptions

- A1: User has sufficient knowledge for initial query.

- A2: Relevance prototypes are “well-behaved”.
  - Term distribution in relevant documents will be similar
  - Term distribution in non-relevant documents will be different from those in relevant documents
    - Either: All relevant documents are tightly clustered around a single prototype.
    - Or: There are different prototypes, but they have significant vocabulary overlap.
    - Similarities between relevant and irrelevant documents are small
Violation of A1

- **User does not have sufficient initial knowledge.**

- **Examples:**
  - Misspellings (Brittany Speers).
  - Cross-language information retrieval (hígado).
  - Mismatch of searcher’s vocabulary vs. collection vocabulary
    - Cosmonaut/astronaut

---

Violation of A2

- **There are several relevance prototypes.**

- **Examples:**
  - Burma/Myanmar
  - Contradictory government policies
  - Pop stars that worked at Burger King

- **Often: instances of a general concept**

- **Good editorial content can address problem**
  - Report on contradictory government policies
Evaluation: Caveat

- True evaluation of usefulness must compare to other methods taking the same amount of time.

- Alternative to relevance feedback: User revises and resubmits query.

- Users may prefer revision/resubmission to having to judge relevance of documents.

- There is no clear evidence that relevance feedback is the “best use” of the user’s time.

Relevance Feedback: Problems

- Long queries are inefficient for typical IR engine.
  - Long response times for user.
  - High cost for retrieval system.
  - Partial solution:
    - Only reweight certain prominent terms
    - Perhaps top 20 by term frequency

- It’s often harder to understand why a particular document was retrieved after applying RF
Relevance Feedback: Problems

- Users are often reluctant to provide explicit feedback
- Implicit feedback and user session monitoring is a better solution
- RF works best when relevant documents form a cluster
- In general negative feedback doesn’t hold a significant improvement

Blind relevance feedback

- Given the initial query search results...
  - a few examples are taken from the top of this rank and the new query is formulated with these positive examples.
- Other strategies can be used to automatically select “possibly” relevant documents
How do we augment the user query?

- **Manual thesaurus**
  - E.g. MedLine: physician, syn: doc, doctor, MD, medico
  - Can be query rather than just synonyms
- **Global Analysis**: (static; of all documents in collection)
  - Automatically derived thesaurus
    - (co-occurrence statistics)
  - Refinements based on query log mining
    - Common on the web
- **Local Analysis**: (dynamic)
  - Analysis of documents in result set

---

Query assist

Would you expect such a feature to increase the query volume at a search engine?
Linguistic thesaurus-based query expansion

- Find synonyms and other morphological forms
- WordNet provides natural language based expansions
  - [http://wordnet.princeton.edu/](http://wordnet.princeton.edu/)

```csharp
public void getSynonyms(IDictionary dict)
{
  // look up first sense of the word "dog"
  IIndexWord idxWord = dict.getIndexWord("dog", POS.NOUN);
  IWordID wordID = idxWord.getIdentity().get(0); // 1st meaning
  IWord word = dict.getWord(wordID);
  ISynset synset = word.getSynset();
  // iterate over words associated with the synset
  for(IWord w : synset.getWords())
  {
    System.out.println(w.getLemmas());
  }
}
```

Xu, J. and Croft, W. B., "Query expansion using local and global document analysis". ACM SIGIR 1996.

Manual thesaurus-based query expansion

- For each term, \( t \), in a query, expand the query with synonyms and related words of \( t \) from the thesaurus
  - feline → feline cat
- May weight added terms less than original query terms.
- Generally increases recall
- Widely used in many science/engineering fields
- May significantly decrease precision, particularly with ambiguous terms.
  - "interest rate" → "interest rate fascinate evaluate"
- There is a high cost of manually producing a thesaurus
  - And for updating it for scientific changes
Automatic Thesaurus Generation

- Attempt to generate a thesaurus automatically by analyzing the collection of documents
- Fundamental notion: similarity between two words
  - Definition 1: Two words are similar if they co-occur with similar words.
  - Definition 2: Two words are similar if they occur in a given grammatical relation with the same words.
- You can harvest, peel, eat, prepare, etc. apples and pears, so apples and pears must be similar.
- Co-occurrence based is more robust, grammatical relations are more accurate.

Co-occurrence Thesaurus

- Simplest way to compute one is based on term-term similarities in $C = AA^T$ where $A$ is term-document matrix.
- $w_{ij} = (\text{normalized})$ weight for $(t_i, d_j)$
- For each $t_i$, pick terms with high values in $C$
Example: Automatic Thesaurus Generation

<table>
<thead>
<tr>
<th>word</th>
<th>ten nearest neighbors</th>
</tr>
</thead>
<tbody>
<tr>
<td>absolutely</td>
<td>absurd whatsoever totally exactly nothing</td>
</tr>
<tr>
<td>bottomed</td>
<td>dip copper drops topped slide trimmed slug</td>
</tr>
<tr>
<td>captivating</td>
<td>shimmer stunningly superbly plucky witty</td>
</tr>
<tr>
<td>doghouse</td>
<td>dog porch crawling beside downstairs gazec</td>
</tr>
<tr>
<td>Makeup</td>
<td>repellent lotion glossy sunscreen Skin gel p</td>
</tr>
<tr>
<td>mediating</td>
<td>reconciliation negotiate cease conciliation p</td>
</tr>
<tr>
<td>keeping</td>
<td>hoping bring wiping could some would other</td>
</tr>
<tr>
<td>lithographs</td>
<td>drawings Picasso Dali sculptures Gauiom</td>
</tr>
<tr>
<td>pathogens</td>
<td>toxins bacteria organisms bacterial parasite</td>
</tr>
<tr>
<td>senses</td>
<td>grasp psyche truly clumsy naive innate awl</td>
</tr>
</tbody>
</table>

Automatic Thesaurus Generation Discussion

- **Quality of associations is usually a problem.**
- **Term ambiguity may introduce irrelevant statistically correlated terms.**
  - “Apple computer” → “Apple red fruit computer”
- **Problems:**
  - False positives: Words deemed similar that are not
  - False negatives: Words deemed dissimilar that are similar
- **Since terms are highly correlated anyway, expansion may not retrieve many additional documents.**
Summary

- **Relevance feedback**
  - Positive and negative feedback
  - Pseudo-relevance (blind) feedback
  - Explicit vs implicit feedback

- **Query expansion**
  - Manual thesaurus
    - Dictionary based
  - Automatic thesaurus
    - Statistical analysis of words co-occurrences in data