Information Retrieval
(Relevance Feedback & Query Expansion)

Fabio Aiolli
http://www.math.unipd.it/~aiolli

Dipartimento di Matematica Pura ed Applicata
Università di Padova

Anno Accademico 2009/2010

Relevance feedback and query expansion

- Goal: To refine the answer set by involving the user in the retrieval process (feedback/interaction)

- Local Methods (adjust the user queries)
  - Relevance feedback
  - Pseudo (or Blind) Relevance Feedback
  - (Global) indirect Relevance Feedback
- Global Methods (independent of the queries and results)
  - Query expansion/reformulation with a thesaurus (WordNet)
  - Query expansion via automatic thesaurus generation
  - Other techniques (spelling correction,…)

Relevance feedback

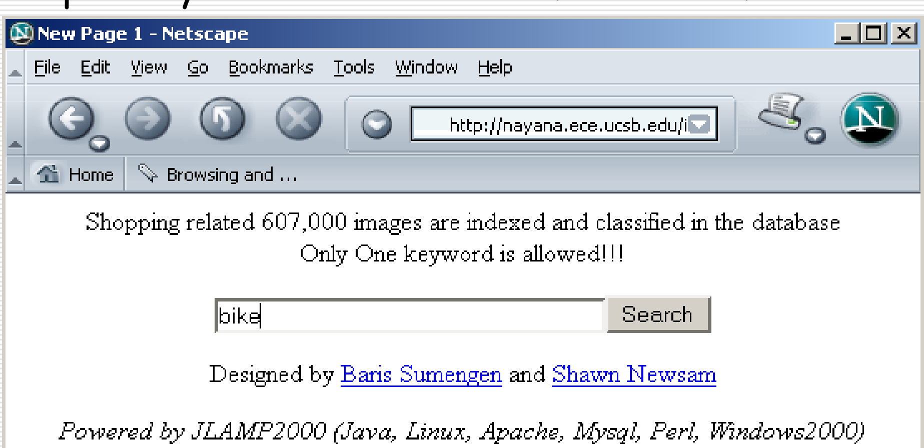
- Basic Procedure of RF
  1. The user issues a simple query
  2. The system returns an initial set of retrieval results
  3. The user marks some of these documents as relevant/irrelevant
  4. The system computes a better representation of the information need based on this feedback
  5. The system displays a revised set of results

  Repeat the procedure one or more times.

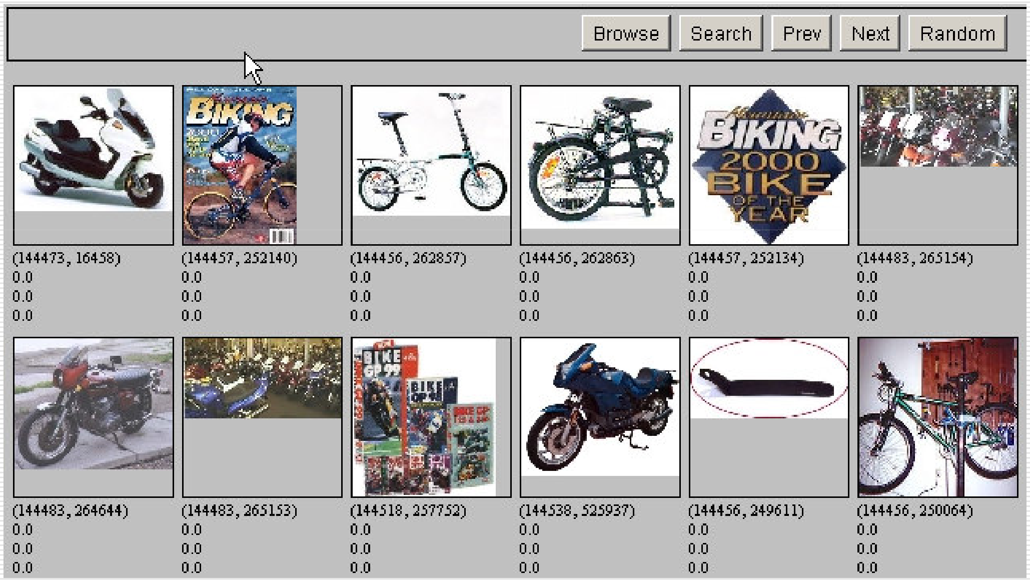
- This process helps the user to focalize its own information need as well.

Relevance Feedback: Example

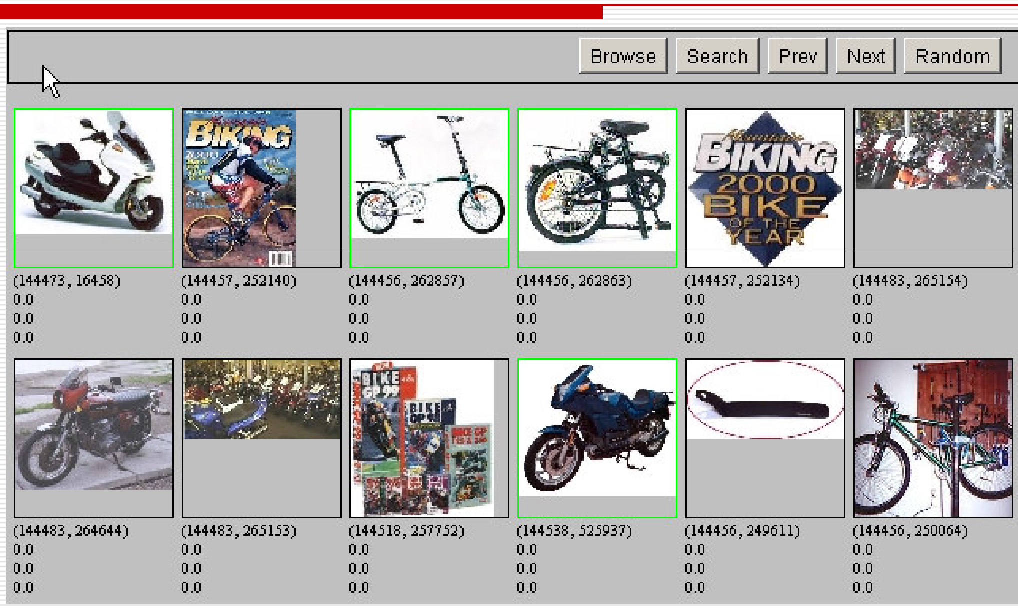
- Image search engine
  http://nayana.ece.ucsb.edu/imsearch/imsearch.html



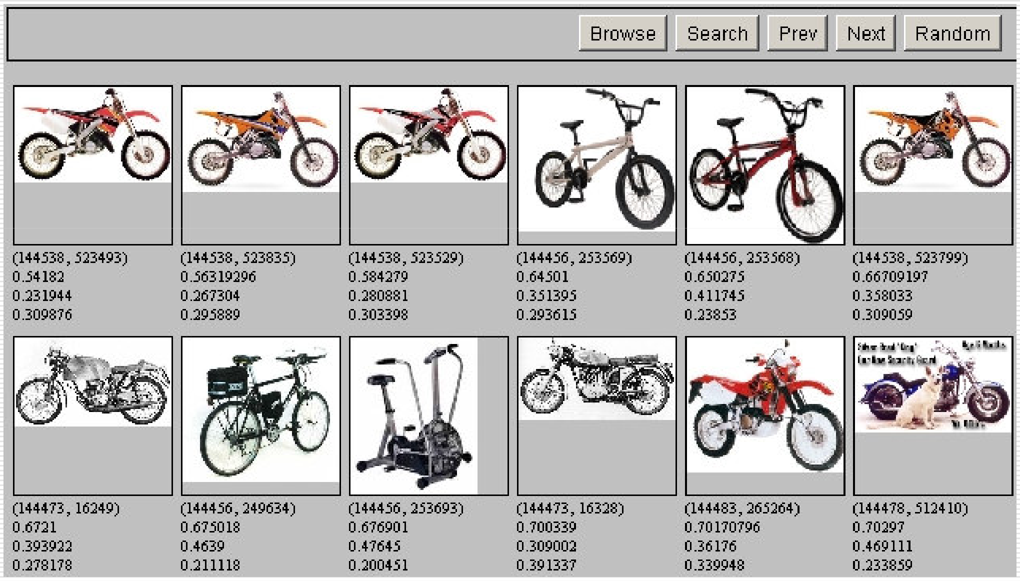
Results for Initial Query



Relevance Feedback step



Results after Relevance Feedback



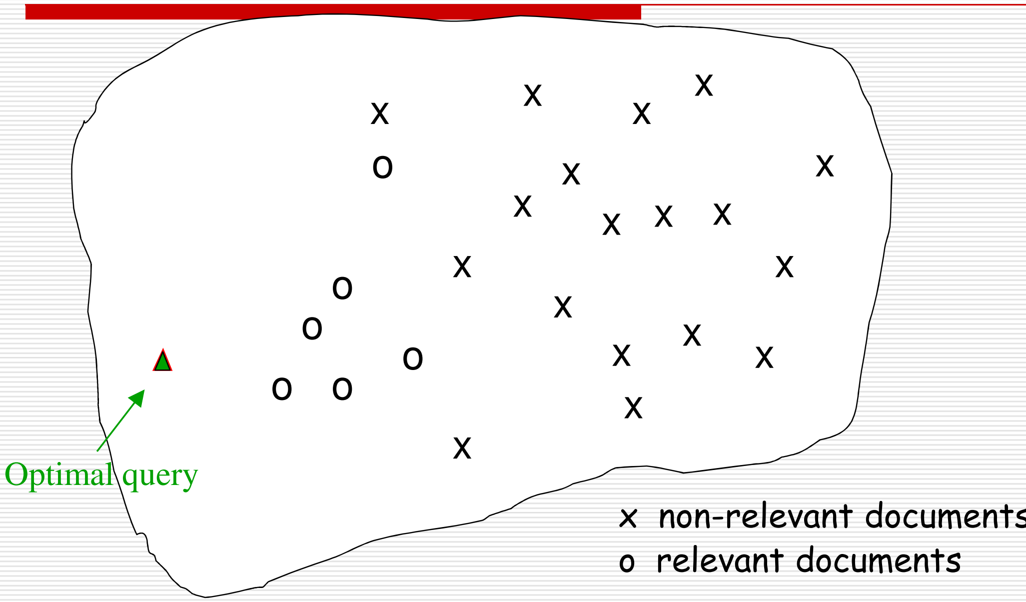
Rocchio Algorithm

- The Rocchio algorithm incorporates relevance feedback information into the vector space model.
- Want to maximize $sim(Q, C_r) - sim(Q, C_{nr})$
- The optimal query vector for separating relevant and non-relevant documents (with cosine sim.):

$$\vec{Q}_{opt} = \frac{1}{|C_r|} \sum_{\vec{d}_j \in C_r} \vec{d}_j - \frac{1}{N - |C_r|} \sum_{\vec{d}_j \notin C_r} \vec{d}_j$$

- Q_{opt} = optimal query; C_r = set of rel. doc vectors; N = collection size
- Unrealistic: we don't know relevant documents.

The Theoretically Best Query



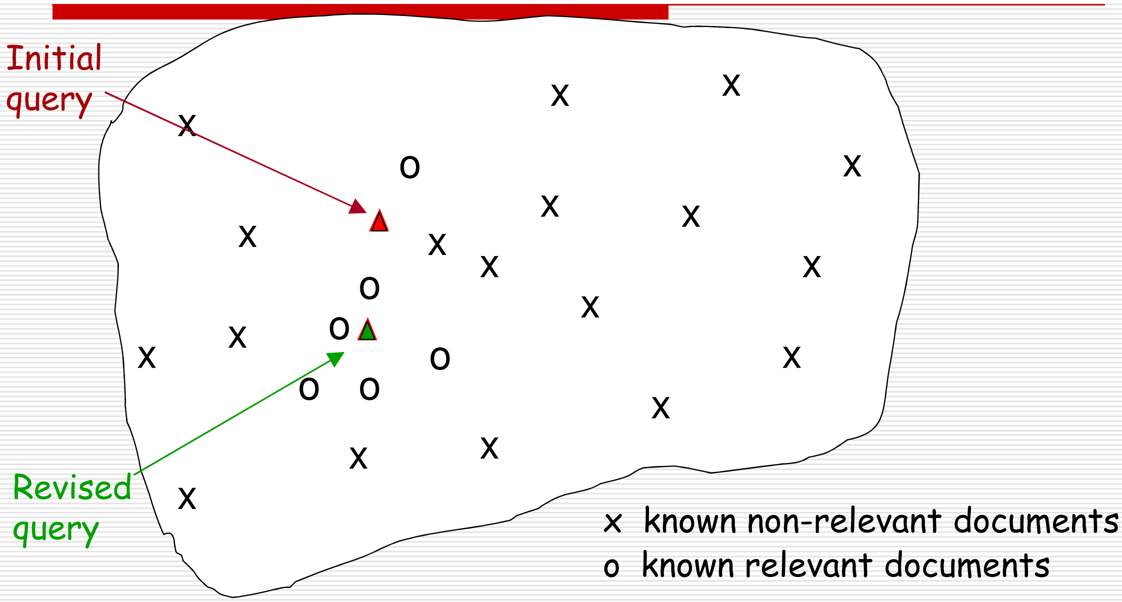
Rocchio 1971 Algorithm (SMART)

- Used in practice:

$$\vec{q}_m = \alpha \vec{q}_0 + \beta \frac{1}{|D_r|} \sum_{\vec{d}_j \in D_r} \vec{d}_j - \gamma \frac{1}{|D_{nr}|} \sum_{\vec{d}_j \in D_{nr}} \vec{d}_j$$

- q_m = modified query vector; q_0 = original query vector; α, β, γ: weights (hand-chosen or set empirically); D_r = set of known relevant doc vectors; D_{nr} = set of known irrelevant doc vectors
- New query moves toward relevant documents and away from irrelevant documents
- Tradeoff α vs. β and γ : If we have a lot of judged documents, we want a higher β and γ.
- Term weight can go negative
  - Negative term weights are ignored
  - Alternatively, weights can be normalized in [0,1]

Relevance feedback on initial query



Relevance Feedback in vector spaces

- We can modify the query based on relevance feedback and apply standard vector space model.
- Use only the docs that were marked.
- Relevance feedback can improve recall and precision
- Relevance feedback is most useful for increasing *recall* in situations where recall is important

Positive vs Negative Feedback

- Usual choices for parameters α, β nad γ
  - $\beta \gg \gamma$, i.e. greater importance to the docs judged relevant than to the docs judged irrelevant
  - $\gamma \neq 0$, as the docs marked irrelevant are typically near-positive. However, many systems only allow positive feedback (γ=0).
  - $\alpha \neq 0$, in order to prevent overfitting, i.e. the excessive influence of 'noisy' characteristics of the docs marked (ir)relevant on the resulting query
  - Reasonable values can be α=1, β=.75, γ=.15

- The values of the parameters could be made dependent on the iteration, i.e. increasing α and decreasing β and γ (later queries already incorporate the contribution of previous feedback iterations)

Probabilistic relevance feedback

- Rather than reweighting in a vector space…
- If user has told us some relevant and irrelevant documents, then we can proceed to build a classifier, such as a Naive Bayes model:
  - $P(t_k|R) = |\mathbf{D}_{rk}| / |\mathbf{D}_r|$
  - $P(t_k|NR) = (N_k - |\mathbf{D}_{rk}|) / (N - |\mathbf{D}_r|)$
    - t_k = term in document; $\mathbf{D}_{rk}$ = known relevant doc containing t_k; N_k = total number of docs containing t_k
- More on later lectures on probabilistic classification
  - This is effectively another way of changing the (implicit) query term weights
  - But note: the above proposal preserves no memory of the original weights

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, laptop / notebook computer

Violation of A2

- There are several relevance prototypes.
- Examples:
  - Burma/Myanmar/Birmania
  - Pop stars that worked at Burger King
  - Often: instances of a very general concept

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
- Users are often reluctant to provide explicit feedback
- It's often harder to understand why a particular document was retrieved after apply relevance feedback

Why?

Relevance Feedback on the Web

[in 2003: now less major search engines, but same general story]

- Some search engines offer a similar/related pages feature (this is a trivial form of relevance feedback)
  - Google (link-based)
  - Altavista
  - Stanford WebBase
- But some don't because it's hard to explain to average user:
  - Alltheweb
  - msn
  - Yahoo
- Excite initially had true relevance feedback, but abandoned it due to lack of use. Why?

Excite Relevance Feedback

Spink et al. 2000 (about Excite)

- Only about 4% of query sessions from a user used relevance feedback option
  - Expressed as "More like this" link next to each result
- But about 70% of users only looked at first page of results and didn't pursue things further
  - So 4% is about 1/8 of people extending search
- Relevance feedback improved results about 2/3 of the time

Exercise

"Find pages like this one!"

What weight setting for α,β,γ ?

Pseudo Relevance Feedback

- Automatic local analysis
- Pseudo relevance feedback attempts to automate the manual part of relevance feedback.
- Retrieve an initial set of relevant documents.
- *Assume* that top *m* ranked documents are relevant.
- Do relevance feedback

- Mostly works (perhaps better than global analysis!)
  - Found to improve performance in TREC ad-hoc task
  - Danger of query drift

Indirect relevance feedback

- On the web, DirectHit introduced a form of indirect relevance feedback.
- DirectHit ranked documents higher that users look at more often.
  - Clicked on links are assumed likely to be relevant
    - Assuming the displayed summaries are good, etc.
- Globally: Not user or query specific.
- This is the general area of clickstream mining (see Joachims work). Applied in advertisment ranking for example.

Relevance Feedback Summary

- Relevance feedback has been shown to be very effective at improving relevance of results.
  - Requires enough judged documents, otherwise it's unstable (≥ 5 recommended)
  - Requires queries for which the set of relevant documents is medium to large
- Full relevance feedback is painful for the user.
- Full relevance feedback is not very efficient in most IR systems.
- Other types of interactive retrieval may improve relevance by as much with less work.

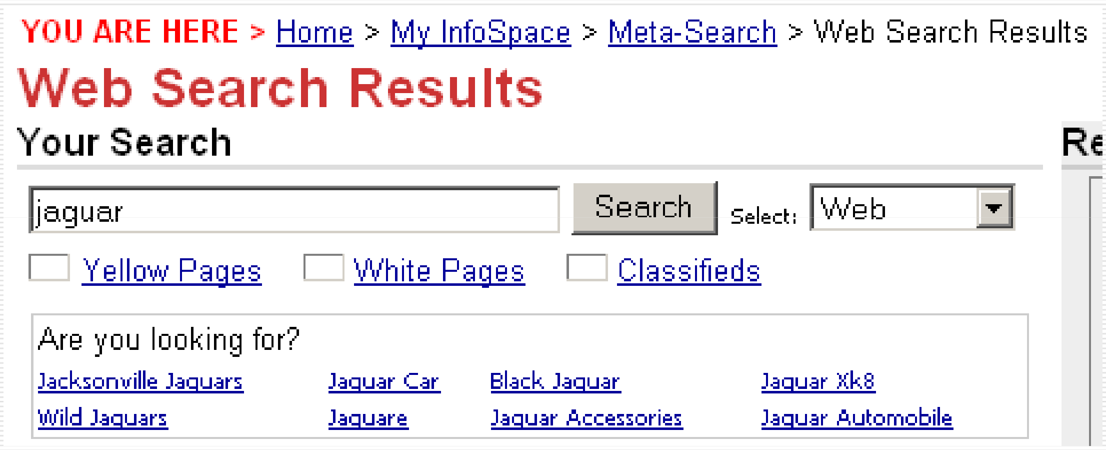
Query Reformulation: Vocabulary Tools

- Feedback
  - Information about stop lists, stemming, etc.
  - Numbers of hits on each term or phrase
- Suggestions
  - Thesaurus
  - Controlled vocabulary
  - Browse lists of terms in the inverted index

Query Expansion

- In relevance feedback, users give additional input (relevant/non-relevant) on documents, which is used to reweight terms in the documents
- In query expansion, users give additional input (good/bad search term) on words or phrases.

Query Expansion: Example

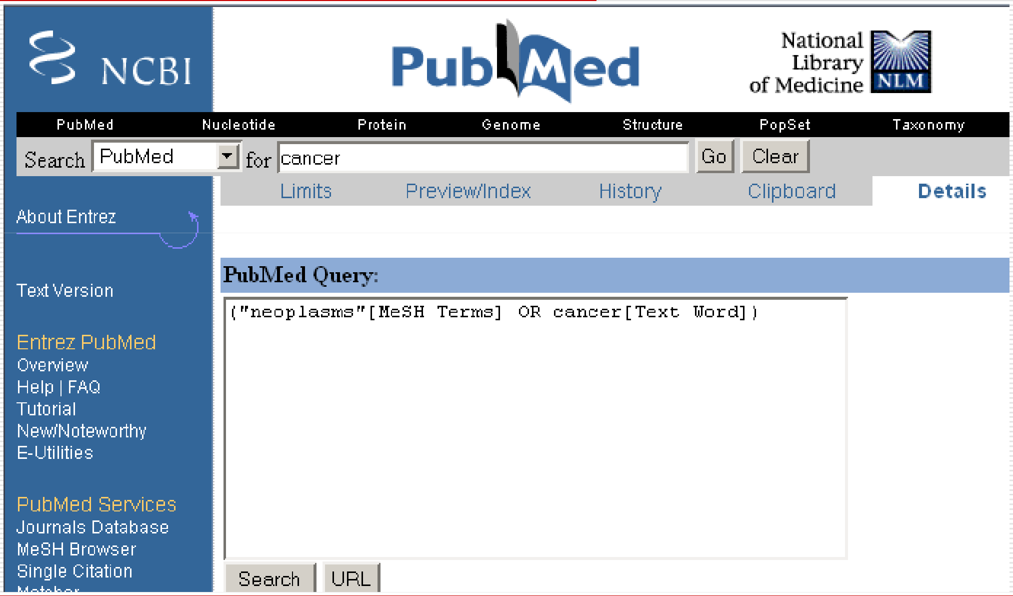


Also: see www.altavista.com, www.teoma.com

Types of Query Expansion

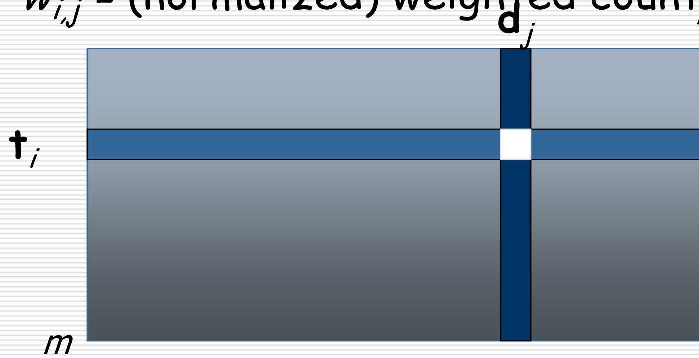
- Global Analysis: Thesaurus-based
  - Controlled vocabulary
    - Maintained by editors (e.g., medline, DD system)
  - Manual thesaurus
    - E.g. MedLine: physician, syn: doc, doctor, MD, medico
  - Automatically derived thesaurus
    - (co-occurrence statistics)
  - Refinements based on query log mining
    - Common on the web
- Local Analysis:
  - Analysis of documents in result set

Controlled Vocabulary



Co-occurrence Thesaurus

- Simplest way to compute one is based on term-term similarities in $C = AA^T$ where A is term-document matrix.
- $w_{i,j}$ = (normalized) weighted count$_n$(t_i, $\mathbf{d}_j$)



With integer counts – what do you get for a boolean cooccurrence matrix?

Automatic Thesaurus Generation Example

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

Query Expansion: Summary

- Query expansion is often effective in increasing recall.
  - Not always with general thesauri
  - Fairly successful for subject-specific collections
- In most cases, precision is decreased, often significantly.
- Overall, not as useful as relevance feedback; *may* be as good as pseudo-relevance feedback