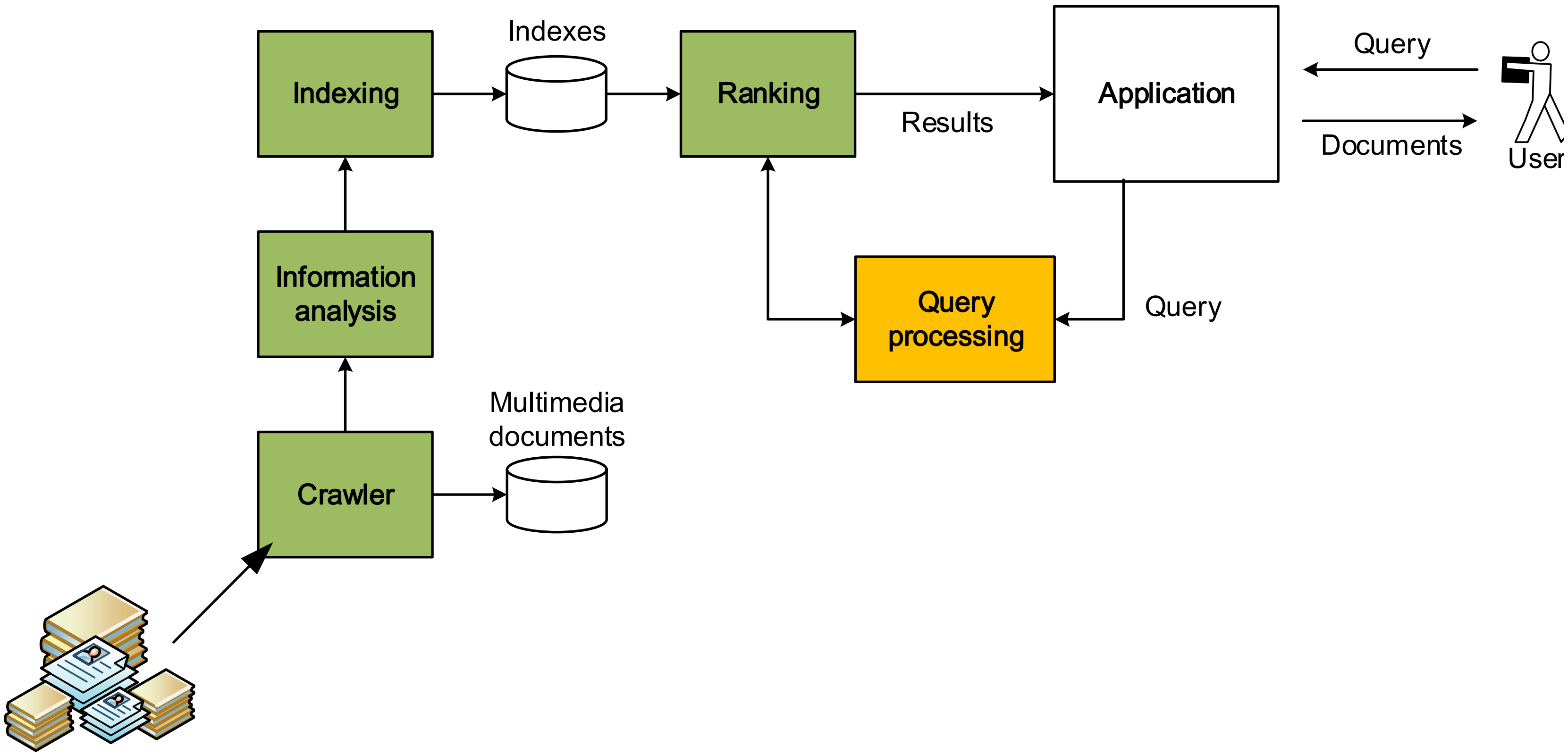
Query Processing

Relevance feedback; query expansion;

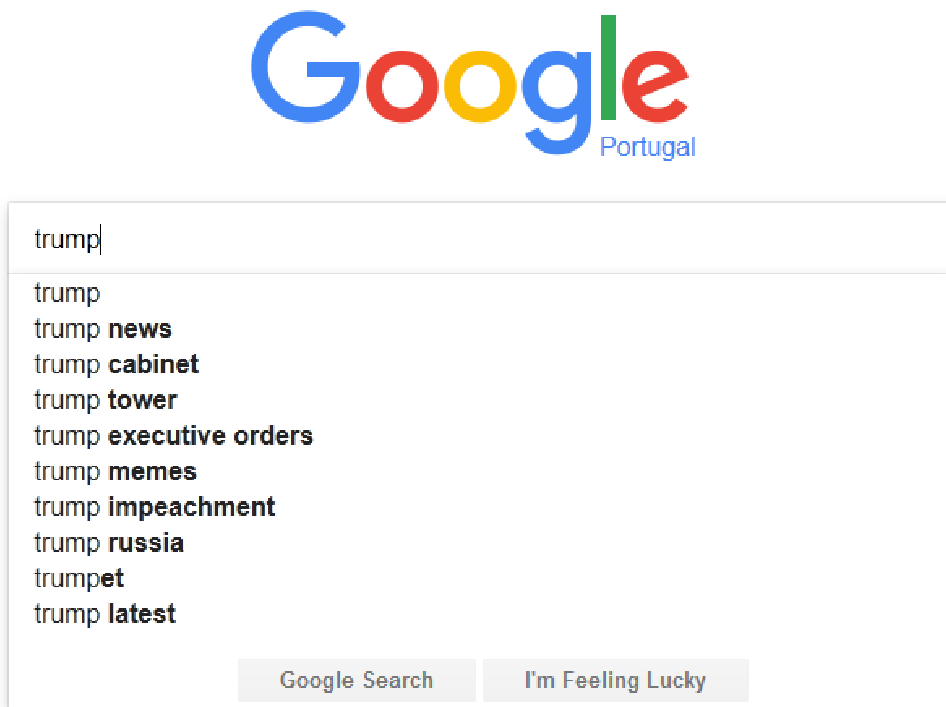
Web Search

Overview

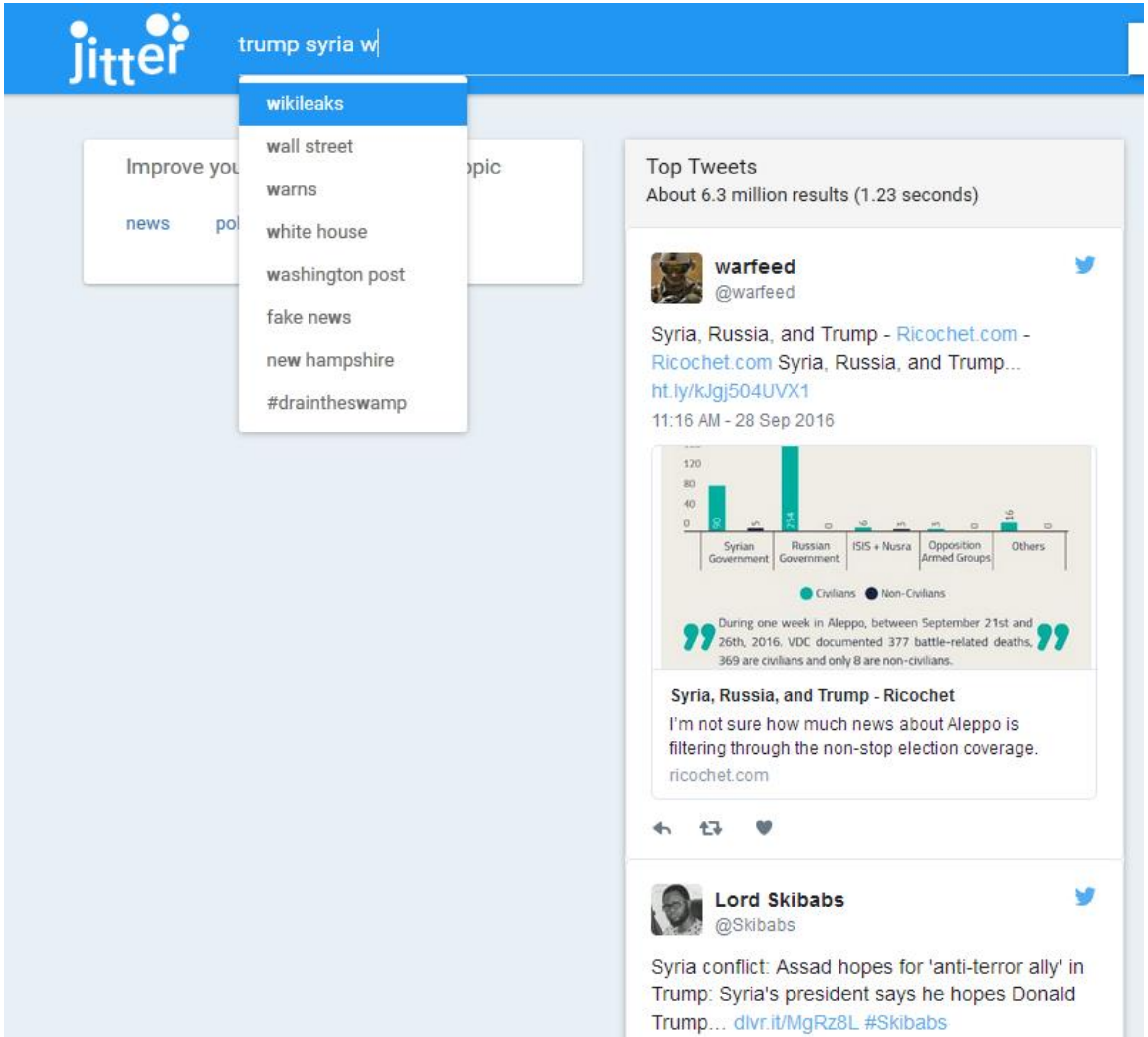


Indexing
Indexes
Ranking
Application
Query
Documents
User
Results
Information analysis
Query processing
Query
Multimedia documents
Crawler

Query assist



Query assist



How can we revise the user query to improve search results?

How do we augment the user query?

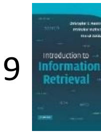
- Local analysis (relevance feedback)
  - Based on the query-related documents (initial search results)

- Global analysis (statistical query expansion)
  - Automatically derived thesaurus from the full collection
  - Refinements based on query log mining

- Manual expansion (thesaurus query expansion)
  - Linguistic thesaurus: e.g. MedLine: physician, syn: doc, doctor, MD, medico
  - Can be query rather than just synonyms

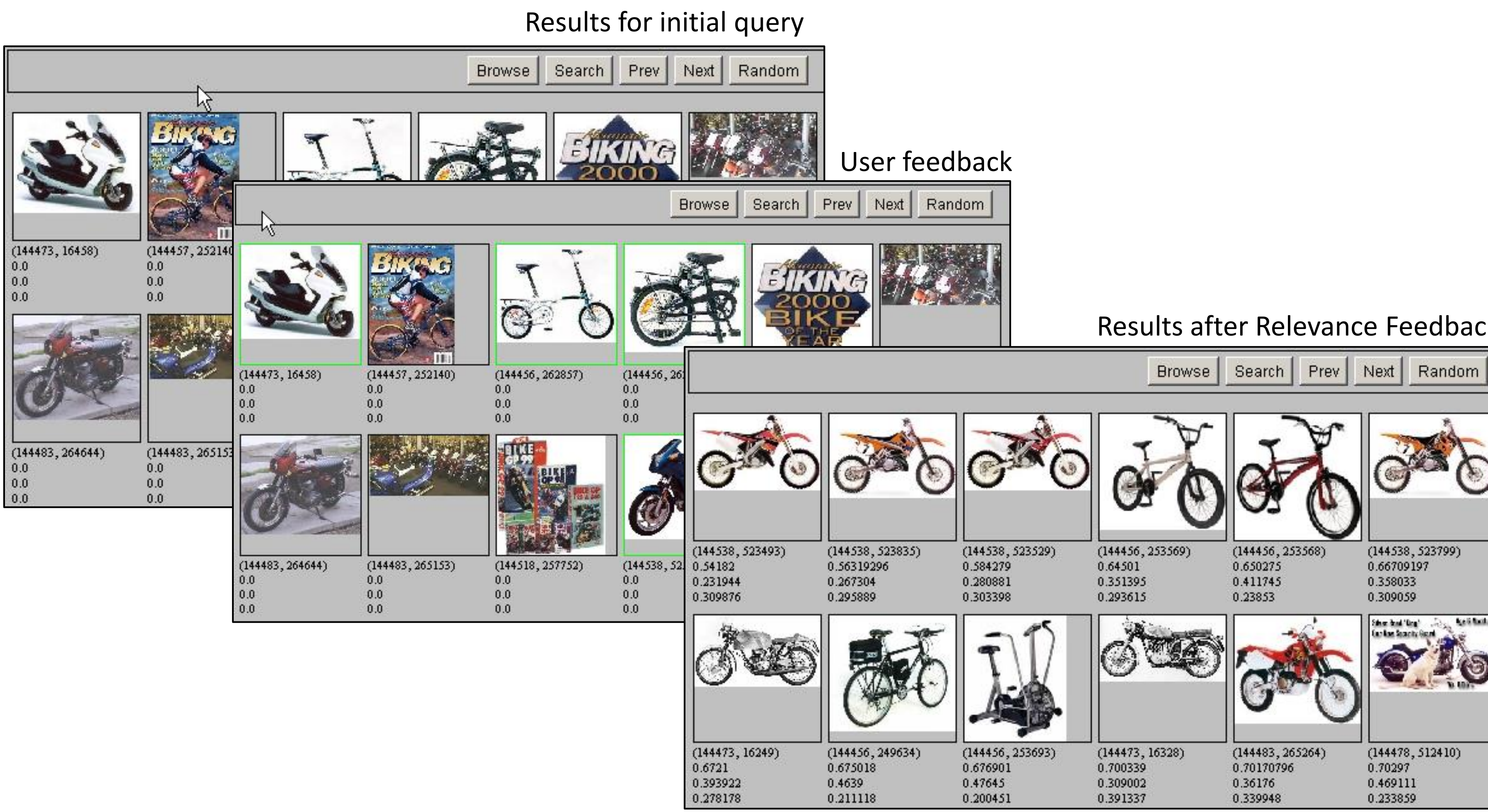
Relevance feedback

Chapter 9



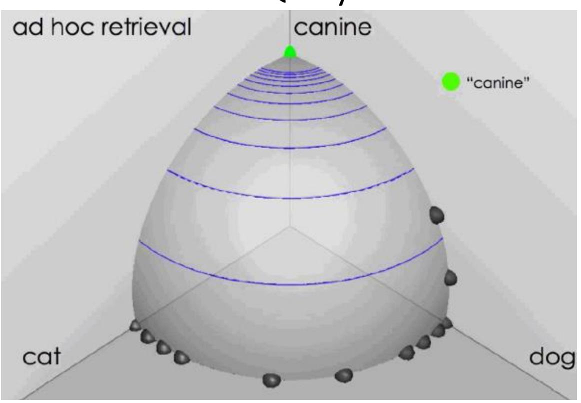
- Given the initial search results, the user marks some documents as important or non-important.
  - This information is used for a second search iteration where these examples are used to refine the results
- The characteristics of the positive examples are used to boost documents with similar characteristics
- The characteristics of the negative examples are used to penalize documents with similar characteristics

Example: UX perspective

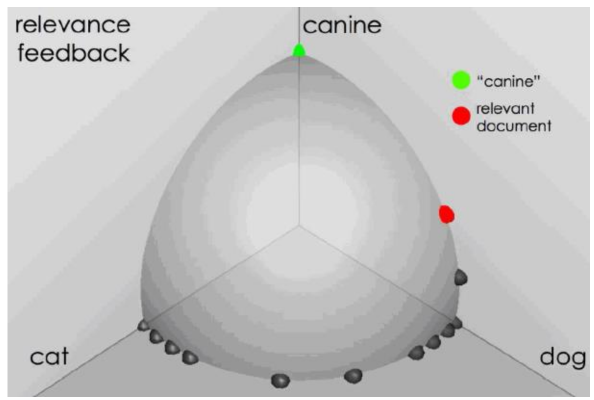


Example: geometric perspective

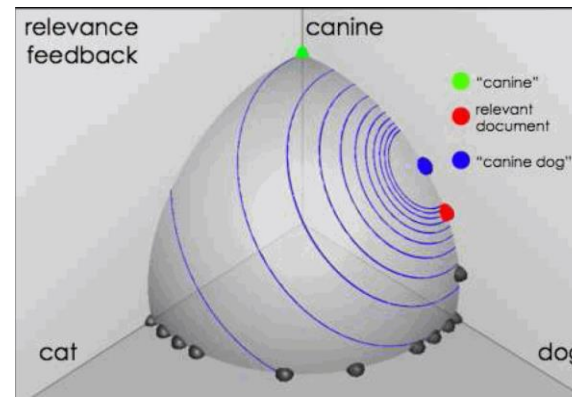
Results for Initial Query



User feedback



Results after Relevance Feedback



Key concept: Centroid

- The <u>centroid</u> is the center of mass of a set of points
  - Recall that we represent documents as points in a high-dimensional space

- The centroid of a set of documents C is defined as:

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

Rocchio algorithm

- The Rocchio algorithm uses the vector space model to pick a relevance fed-back query
  - Rocchio seeks the query q_{opt} that maximizes

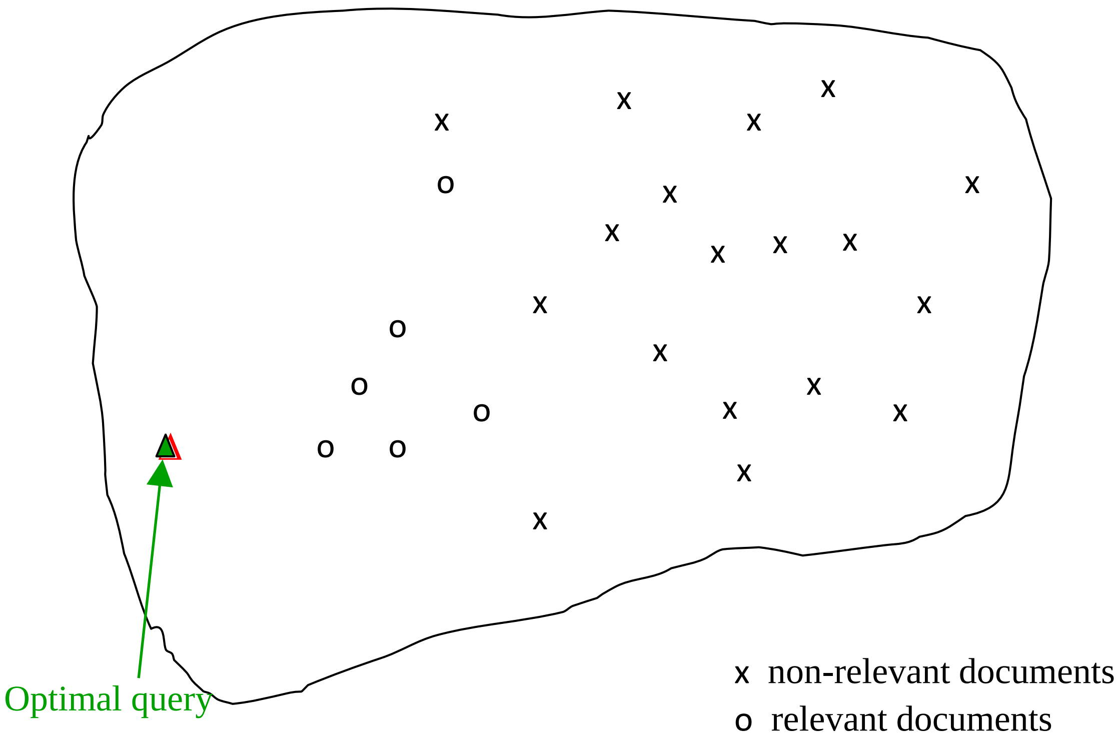
$$\vec{q}_{opt} = \arg\max_{\vec{q}} [\cos(\vec{q}, \vec{\mu}(C_r)) - \cos(\vec{q}, \vec{\mu}(C_{nr}))]$$

- Tries to separate documents marked as relevant and non-relevant

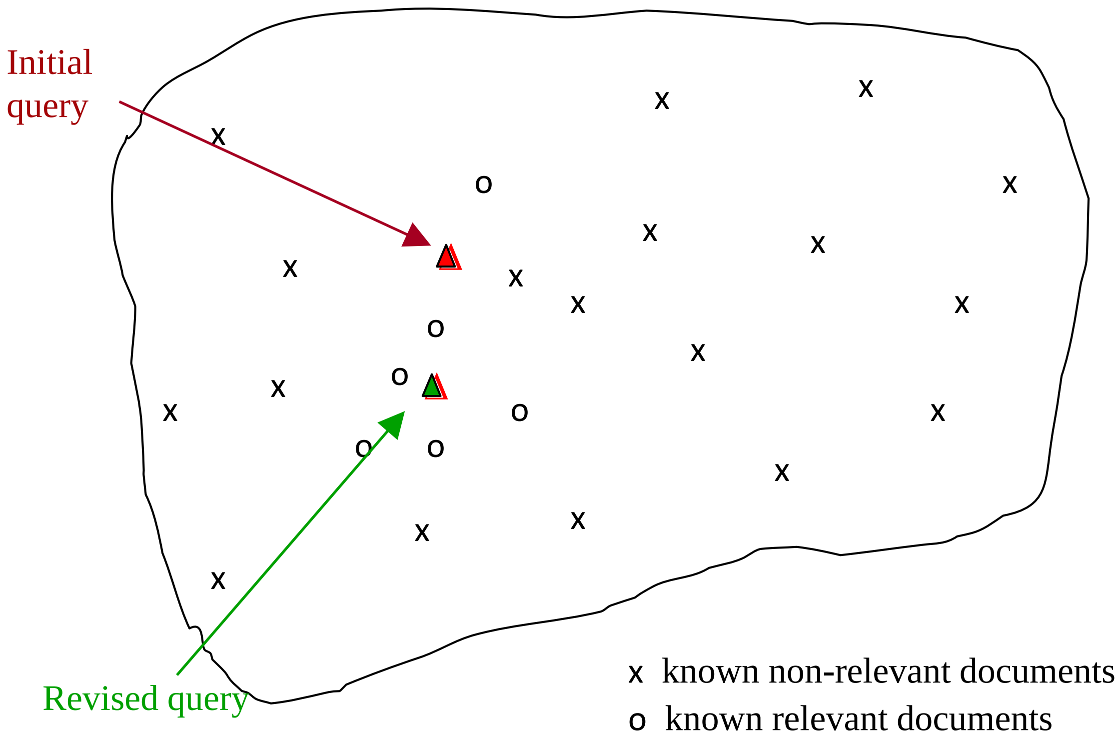
$$\vec{q}_{opt} = \frac{1}{|C_r|} \sum_{\vec{d}_j \in C_r} \vec{d}_j - \frac{1}{|C_{nr}|} \sum_{\vec{d}_j \notin C_r} \vec{d}_j$$

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

The theoretically best query



Relevance feedback on initial query



Rocchio 1971 Algorithm (SMART)

- Used in practice:

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

  - 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; α,β,γ: weights (hand-chosen or set empirically)

- The new query moves toward relevant documents and away from irrelevant documents

Subtleties to note

- Tradeoff α vs. β/γ : If we have a lot of judged documents, we want a higher β/γ.
- Some weights in query vector can go negative
  - Negative term weights are ignored (set to 0)

Google A/B testing of relevance feedback

Relevance feedback: Why is it not used?

- 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

Relevance feedback: Assumptions

Sec. 9.1.3

- 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

Sec. 9.1.3

- 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

Sec. 9.1.3

- 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.

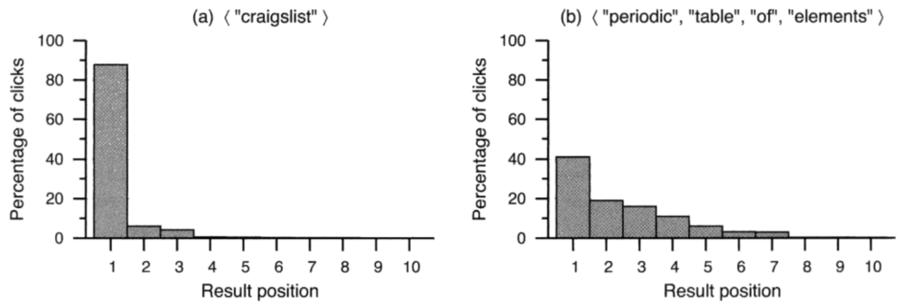


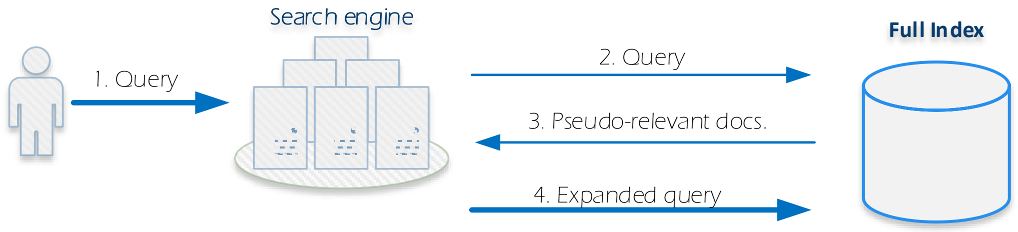
Figure 15.5 Clickthrough curve for a typical navigational query (⟨"craigslist"⟩) and a typical informational query (⟨"periodic", "table", "of", "elements"⟩).

- There is no clear evidence that relevance feedback is the "best use" of the user's time

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

Pseudo-relevance feedback

- Given the initial query search results…
  - a few examples are taken from the top of this rank and a new query is formulated with these positive examples.



- It is important to chose the right number of documents and the terms to expand the query

Pseudo-relevant feedback

Sec. 9.1.1

- The most frequent terms of all top documents are considered the pseudo-relevant terms:

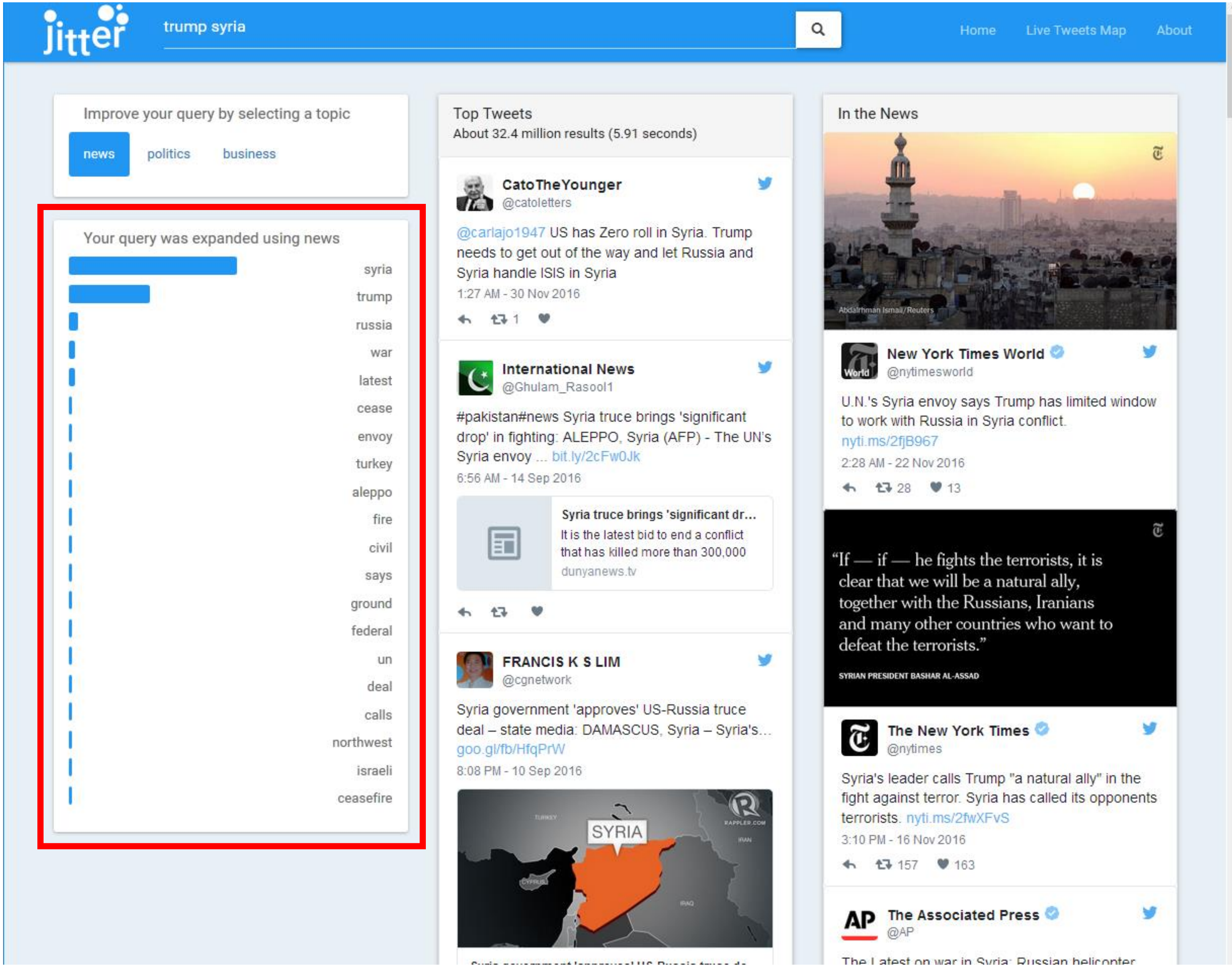
$$topDocTerms = \sum_{i=1}^{\#topDocs} d_{retDocId(q_0,i)}$$

$$prfterms_i = \begin{cases} topDocTerms_i & topDocTerms_i < th \\ 0 & topDocTerms_i < th \end{cases}$$

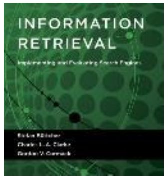
$$, s.t. \ \|prfterms\|_0 = \#topterms$$

- The expanded queries then become: $q = \gamma \cdot q_0 + (1 - \gamma) \cdot prfterms$

- Other strategies can be thought to automatically select “possibly” relevant documents


jitter
trump syria
Home
Live Tweets Map
About
Improve your query by selecting a topic
news
politics
business
Your query was expanded using news
syria
trump
russia
war
latest
cease
envoy
turkey
aleppo
fire
civil
says
ground
federal
un
deal
calls
northwest
israeli
ceasefire
Top Tweets
About 32.4 million results (5.91 seconds)
CatoTheYounger
@catoletters
@carlajo1947 US has Zero roll in Syria. Trump needs to get out of the way and let Russia and Syria handle ISIS in Syria
1:27 AM - 30 Nov 2016
International News
@Ghulam_Rasool1
#pakistan#news Syria truce brings 'significant drop' in fighting: ALEPPO, Syria (AFP) - The UN's Syria envoy ... bit.ly/2cFw0Jk
6:56 AM - 14 Sep 2016
Syria truce brings 'significant dr...
It is the latest bid to end a conflict that has killed more than 300,000
dunyanews.tv
FRANCIS K S LIM
@cgnetwork
Syria government 'approves' US-Russia truce deal – state media: DAMASCUS, Syria – Syria's...
goo.gl/fb/HfqPrW
8:08 PM - 10 Sep 2016
SYRIA
In the News
Abdalrhman Ismail/Reuters
New York Times World
@nytimesworld
U.N.'s Syria envoy says Trump has limited window to work with Russia in Syria conflict.
nyti.ms/2fjB967
2:28 AM - 22 Nov 2016
28
13
"If — if — he fights the terrorists, it is clear that we will be a natural ally, together with the Russians, Iranians and many other countries who want to defeat the terrorists."
SYRIAN PRESIDENT BASHAR AL-ASSAD
The New York Times
@nytimes
Syria's leader calls Trump "a natural ally" in the fight against terror. Syria has called its opponents terrorists. nyti.ms/2fwXFvS
3:10 PM - 16 Nov 2016
157
163
The Associated Press
@AP

Experimental comparison



	TREC45				Gov2			
	1998		1999		2004		2005	
Method	**P@10**	**MAP**	**P@10**	**MAP**	**P@10**	**MAP**	**P@10**	**MAP**
Cosine TF-IDF	0.264	0.126	0.252	0.135	0.120	0.060	0.194	0.092
Proximity	0.396	0.124	0.370	0.146	0.425	0.173	0.562	0.23
No length norm. (rawTF)	0.266	0.106	0.240	0.120	0.298	0.093	0.282	0.097
D: rawTF+ noIDF Q: IDF	0.342	0.132	0.328	0.154	0.400	0.144	0.466	0.151
Binary	0.256	0.141	0.224	0.148	0.069	0.050	0.106	0.083
2-Poisson	0.402	0.177	0.406	0.207	0.418	0.171	0.538	0.207
BM25	0.424	0.178	0.440	0.205	0.471	0.243	0.534	0.277
LMD	0.450	0.193	0.428	0.226	0.484	0.244	0.580	0.293
BM25F					0.482	0.242	0.544	0.277
BM25+PRF	0.452	0.239	0.454	0.249	0.567	0.277	0.588	0.314
RRF	0.462	0.215	0.464	0.252	0.543	0.297	0.570	0.352
LR			0.446	0.266			0.588	0.309
RankSVM			0.420	0.234			0.556	0.268

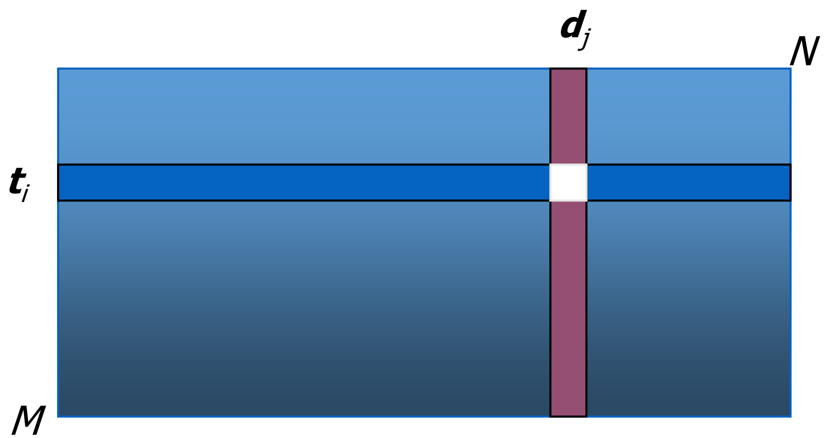
How do we augment the user query?

- Local analysis (relevance feedback)
  - Based on the query-related documents (initial search results)
- Global analysis (statistical query expansion)
  - Automatically derived thesaurus from the full collection
  - Refinements based on query log mining
- Manual expansion (thesaurus query expansion)
  - Linguistic thesaurus: e.g. MedLine: physician, syn: doc, doctor, MD, medico
  - Can be query rather than just synonyms

Co-occurrence thesaurus

Sec. 9.2.3

- 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) weight for $(t_i, \boldsymbol{d}_j)$



What does C contain if A is a term-doc incidence (0/1) matrix?

- For each t_i, pick terms with high values in C

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.

- Co-occurrence based is more robust, grammatical relations are more accurate.

Example: Automatic thesaurus generation

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

If the initial query has 3 terms, the query that "hits" the index may end-up having 30 terms!!!

Retrieval precision improves, but, how is retrieval efficiency affected by this?

How do we augment the user query?

- Local analysis (relevance feedback)
  - Based on the query-related documents (initial search results)
- Global analysis (statistical query expansion)
  - Automatically derived thesaurus from the full collection
  - Refinements based on query log mining
- Manual expansion (thesaurus query expansion)
  - Linguistic thesaurus: e.g. MedLine: physician, syn: doc, doctor, MD, medico
  - Can be query rather than just synonyms

Linguistic thesaurus-based query expansion

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

```
public void getSynonyms(IDictionary dict){

  // look up first sense of the word "dog"
  IIndexWord idxWord = dict.getIndexWord("dog", POS.NOUN);
  IWordID wordID = idxWord.getWordIDs().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.getLemma());
  }
}
```

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 <u>increases recall</u>
  - Widely used in many science/engineering fields
- May significantly <u>decrease precision</u>, particularly with ambiguous terms.
  - "interest rate" $\rightarrow$ "interest rate fascinate evaluate"
  - There is a high cost of manually producing a thesaurus

Summary

- PRF improves top precision and QE improves recall but…
- It's often harder to understand why a particular document was retrieved after applying RF or QE
- 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