PV211: Introduction to Information Retrieval

https://www.fi.muni.cz/~sojka/PV211

IIR 9: Relevance feedback & Query expansion
Handout version

Petr Sojka, Hinrich Schütze et al.

Faculty of Informatics, Masaryk University, Brno
Center for Information and Language Processing, University of Munich

2021-04-19

(compiled on 2021-06-22 16:00:40)

Overview

Take-away today

- Interactive relevance feedback: improve initial retrieval results by telling the IR system which docs are relevant / non-relevant
- Best known relevance feedback method: Rocchio feedback
- Query expansion: improve retrieval results by adding synonyms / related terms to the query
    - Sources for related terms: Manual thesauri, automatic thesauri, query logs

How can we improve recall in search?

- Main topic today: two ways of improving recall: relevance feedback and query expansion
- As an example consider query q: [aircraft] . . .
- . . . and document d containing "plane", but not containing "aircraft"
- A simple IR system will not return d for q.
- Even if d is the most relevant document for q!
- We want to change this:
  - Return relevant documents even if there is no term match with the (original) query

Recall

- Loose definition of recall in this lecture: "increasing the number of relevant documents returned to user"
- This may actually decrease recall on some measures, e.g., when expanding "jaguar" with "panthera"
  - . . . which eliminates some relevant documents, but increases relevant documents returned on top pages

Options for improving recall

- Local: Do a "local", on-demand analysis for a user query
  - Main local method: relevance feedback
  - Part 1
- Global: Do a global analysis once (e.g., of collection) to produce thesaurus
  - Use thesaurus for query expansion
  - Part 2

Google examples for query expansion

- One that works well
  - *˜flights -flight*
- One that doesn't work so well
  - *˜dogs -dog*

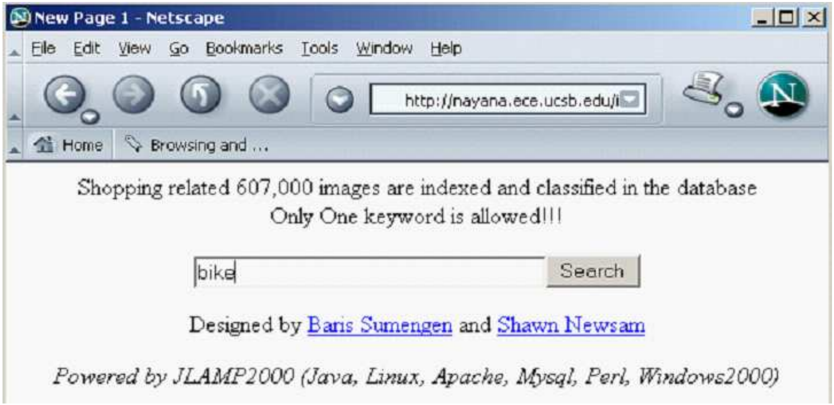
Relevance feedback: Basic idea

- The user issues a (short, simple) query.
- The search engine returns a set of documents.
- User marks some docs as relevant, some as non-relevant.
- Search engine computes a new representation of the information need. Hope: better than the initial query.
- Search engine runs new query and returns new results.
- New results have (hopefully) better recall.
- We can iterate this: several rounds of relevance feedback.
- We will use the term ad hoc retrieval to refer to regular retrieval without relevance feedback.

Relevance feedback: Examples

- We will now look at three different examples of relevance feedback that highlight different aspects of the process.

Relevance Feedback: Example 1


New Page 1 - Netscape
File Edit View Go Bookmarks Tools Window Help
http://nayana.ece.ucsb.edu/i
Home
Browsing and ...
Search

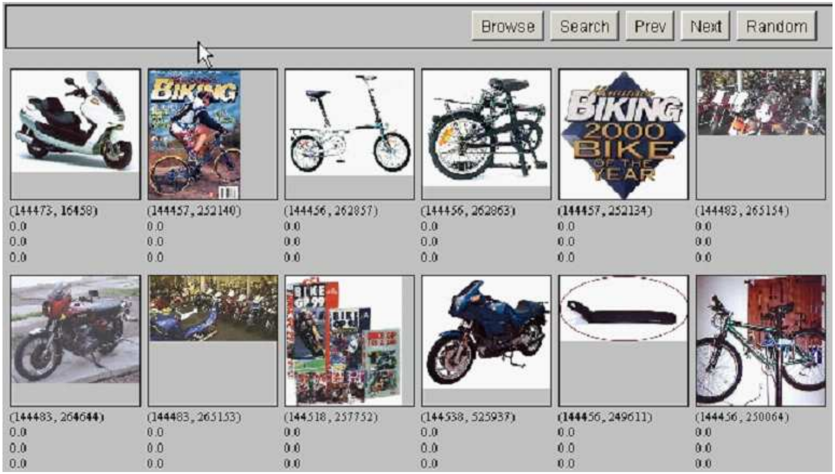
Shopping related 607,000 images are indexed and classified in the database
Only One keyword is allowed!!!

bike

Designed by Baris Sumengen and Shawn Newsam

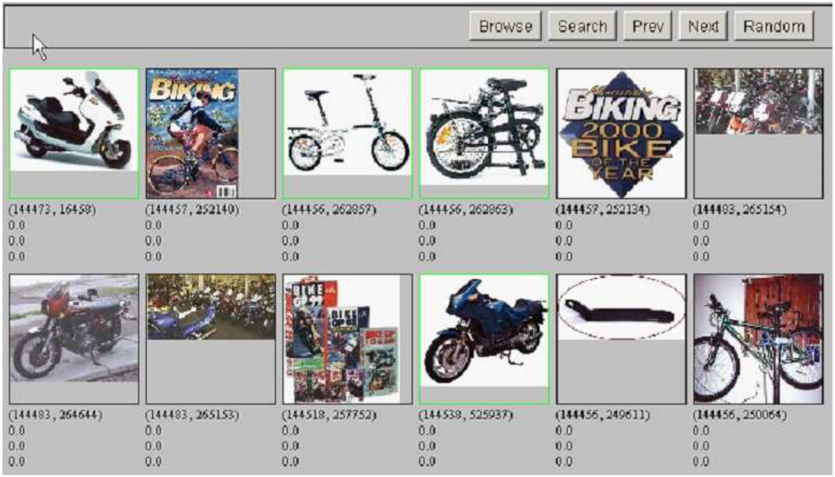
Powered by JLAMP2000 (Java, Linux, Apache, Mysql, Perl, Windows2000)

Results for initial query



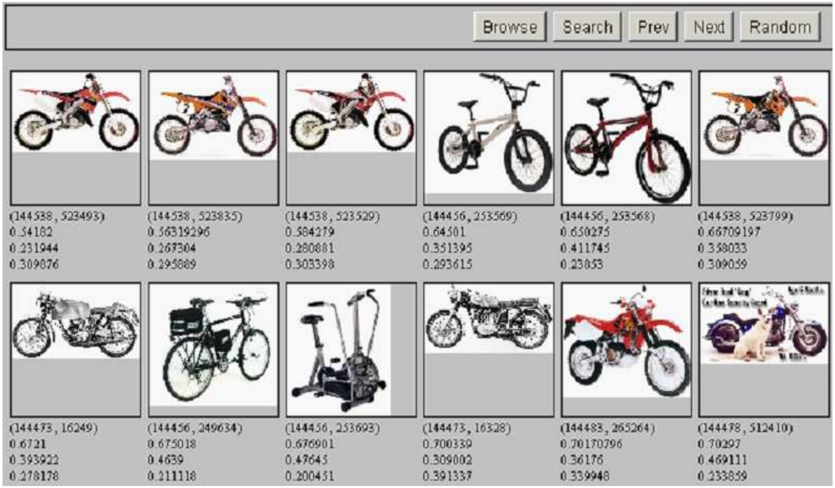
(144473, 16458)	(144457, 252140)	(144456, 262857)	(144456, 262863)	(144457, 252134)	(144483, 265154)
0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0
(144483, 264644)	(144483, 265153)	(144518, 257752)	(144538, 525937)	(144456, 249611)	(144456, 250064)
0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0

User feedback: Select what is relevant



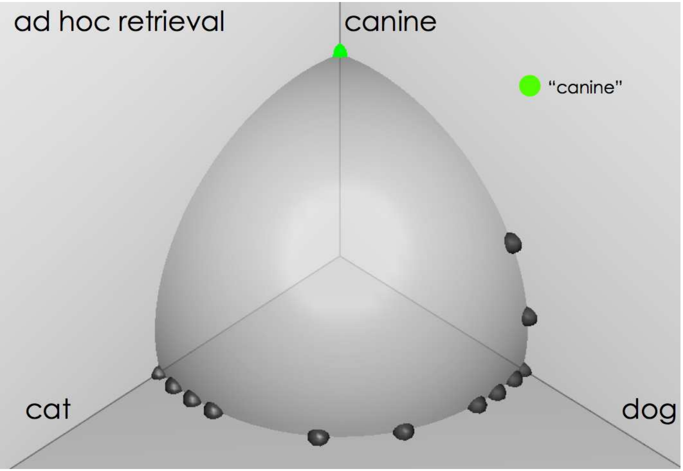
```
(144473, 16458)  (144457, 252140)  (144456, 262857)  (144456, 262863)  (144457, 252134)  (144483, 265154)
0.0              0.0               0.0               0.0               0.0               0.0
0.0              0.0               0.0               0.0               0.0               0.0
0.0              0.0               0.0               0.0               0.0               0.0

(144483, 264644) (144483, 265153)  (144518, 257752)  (144538, 525937)  (144456, 249611)  (144456, 250064)
0.0              0.0               0.0               0.0               0.0               0.0
0.0              0.0               0.0               0.0               0.0               0.0
0.0              0.0               0.0               0.0               0.0               0.0
```

Results after relevance feedback



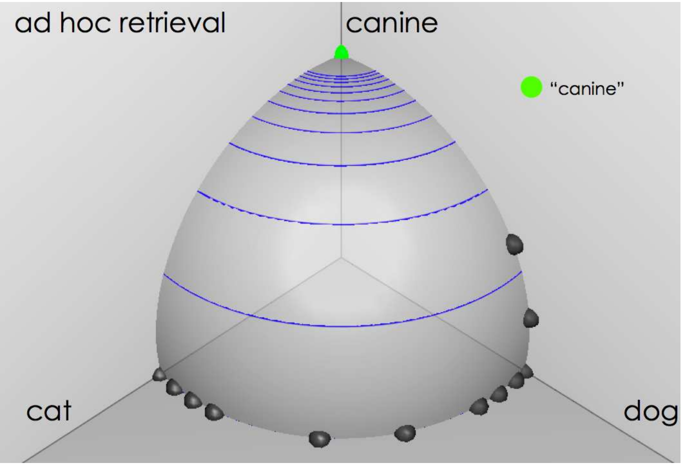
(144538, 523493)	(144538, 523835)	(144538, 523529)	(144456, 253569)	(144456, 253568)	(144538, 523799)
0.54182	0.56319296	0.584279	0.64501	0.650275	0.66709197
0.231944	0.267304	0.280881	0.351395	0.411745	0.358033
0.309876	0.295889	0.303398	0.293615	0.23853	0.309059
(144473, 16249)	(144456, 249634)	(144456, 253693)	(144473, 16328)	(144483, 265264)	(144478, 512410)
0.6721	0.675018	0.676901	0.700339	0.70170796	0.70297
0.393922	0.4639	0.47645	0.309002	0.36176	0.469111
0.278178	0.211118	0.200451	0.391337	0.339948	0.233859

Vector space example: query “canine” (1)



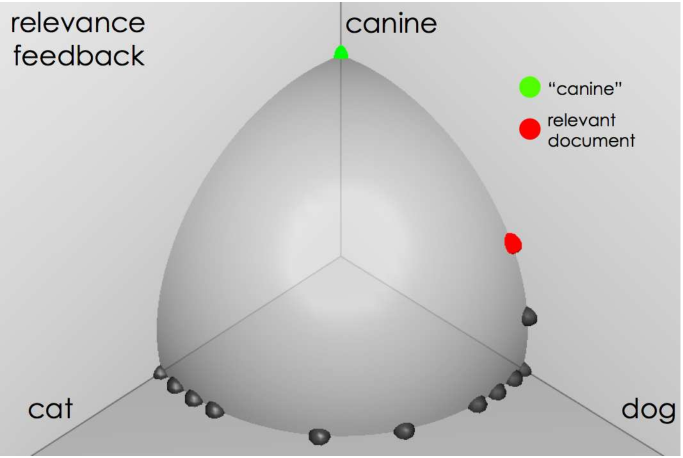
source:
Fernando Díaz

Similarity of docs to query “canine”



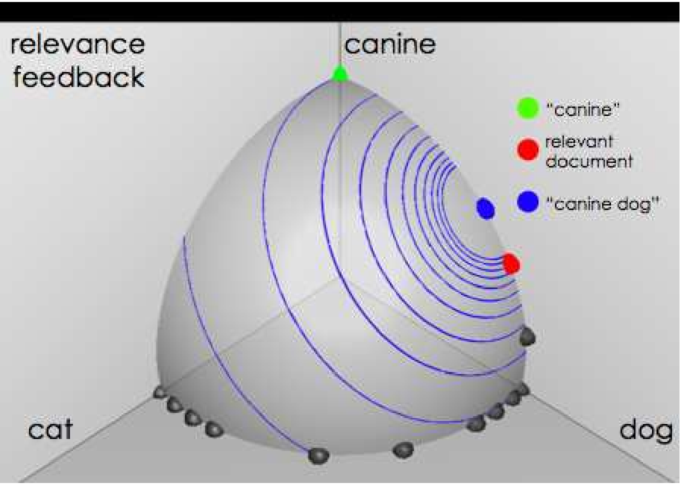
source:
Fernando Díaz

User feedback: Select relevant documents



source:
Fernando Díaz

Results after relevance feedback



source:
Fernando Díaz

Example 3: A real (non-image) example

Initial query: [new space satellite applications]

Results for initial query: (r = rank)

	r		
+	1	0.539	NASA Hasn't Scrapped Imaging Spectrometer
+	2	0.533	NASA Scratches Environment Gear From Satellite Plan
	3	0.528	Science Panel Backs NASA Satellite Plan, But Urges Launches of Smaller Probes
	4	0.526	A NASA Satellite Project Accomplishes Incredible Feat: Staying Within Budget
	5	0.525	Scientist Who Exposed Global Warming Proposes Satellites for Climate Research
	6	0.524	Report Provides Support for the Critics Of Using Big Satellites to Study Climate
	7	0.516	Arianespace Receives Satellite Launch Pact From Telesat Canada
+	8	0.509	Telecommunications Tale of Two Companies

User then marks relevant documents with "+".

Expanded query after relevance feedback

2.074	new	15.106	space
30.816	satellite	5.660	application
5.991	nasa	5.196	eos
4.196	launch	3.972	aster
3.516	instrument	3.446	arianespace
3.004	bundespost	2.806	ss
2.790	rocket	2.053	scientist
2.003	broadcast	1.172	earth
0.836	oil	0.646	measure

Compare to original query: [new space satellite applications]

Results for expanded query (old ranks in parens)

	r		
*	1 (2)	0.513	NASA Scratches Environment Gear From Satellite Plan
*	2 (1)	0.500	NASA Hasn't Scrapped Imaging Spectrometer
	3	0.493	When the Pentagon Launches a Secret Satellite, Space Sleuths Do Some Spy Work of Their Own
	4	0.493	NASA Uses 'Warm' Superconductors For Fast Circuit
*	5 (8)	0.492	Telecommunications Tale of Two Companies
	6	0.491	Soviets May Adapt Parts of SS-20 Missile For Commercial Use
	7	0.490	Gaping Gap: Pentagon Lags in Race To Match the Soviets In Rocket Launchers
	8	0.490	Rescue of Satellite By Space Agency To Cost $90 Million

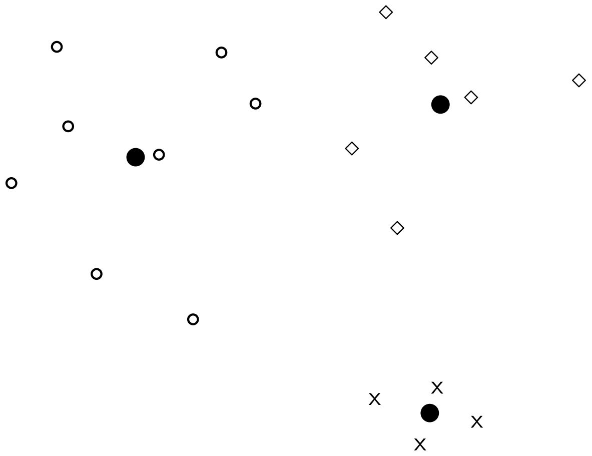
Key concept for relevance feedback: Centroid

- The centroid is the center of mass of a set of points.
- Recall that we represent documents as points in a high-dimensional space.
- Thus: we can compute centroids of documents.
- Definition:

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

where D is a set of documents and $\vec{v}(d) = \vec{d}$ is the vector we use to represent document d.

Centroid: Examples



Rocchio algorithm

- The Rocchio algorithm implements relevance feedback in the vector space model.
- Rocchio chooses the query $\vec{q}_{opt}$ that maximizes

$$\vec{q}_{opt} = \arg\max_{\vec{q}} [\text{sim}(\vec{q}, \mu(D_r)) - \text{sim}(\vec{q}, \mu(D_{nr}))]$$

  D_r: set of relevant docs; D_{nr}: set of nonrelevant docs
- Intent: $\vec{q}_{opt}$ is the vector that separates relevant and non-relevant docs maximally.
- Making some additional assumptions, we can rewrite $\vec{q}_{opt}$ as:

$$\vec{q}_{opt} = \mu(D_r) + [\mu(D_r) - \mu(D_{nr})]$$

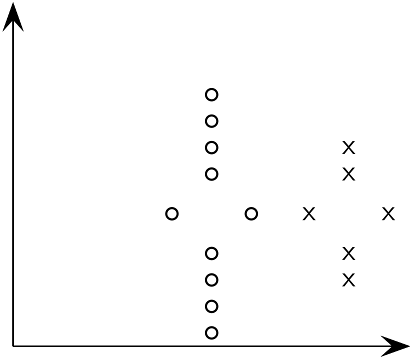
Rocchio algorithm

- The optimal query vector is:

$$\begin{aligned}\vec{q}_{opt} &= \mu(D_r) + [\mu(D_r) - \mu(D_{nr})] \\ &= \frac{1}{|D_r|}\sum_{\vec{d}_j \in D_r} \vec{d}_j + [\frac{1}{|D_r|}\sum_{\vec{d}_j \in D_r} \vec{d}_j - \frac{1}{|D_{nr}|}\sum_{\vec{d}_j \in D_{nr}} \vec{d}_j]\end{aligned}$$

- We move the centroid of the relevant documents by the difference between the two centroids.

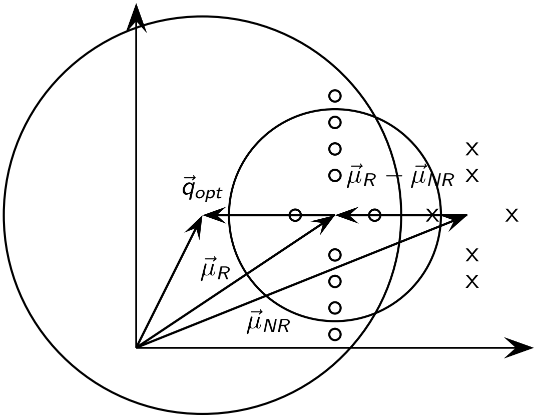
Exercise: Compute Rocchio vector



circles: relevant documents, Xs: nonrelevant documents

compute: $\vec{q}_{opt} = \mu(D_r) + [\mu(D_r) - \mu(D_{nr})]$

Rocchio illustrated



circles: relevant documents, Xs: non-relevant documents $\vec{\mu}_R$: centroid of relevant documents $\vec{\mu}_R$ does not separate relevant/non-relevant. $\vec{\mu}_{NR}$: centroid of non-relevant documents $\vec{\mu}_R - \vec{\mu}_{NR}$: difference vector Add difference vector to $\vec{\mu}_R$ ... ...to get $\vec{q}_{opt}$ $\vec{q}_{opt}$ separates relevant/non-relevant perfectly.

Terminology

- So far, we have used the name Rocchio for the theoretically better motivated original version of Rocchio.
- The implementation that is actually used in most cases is the SMART implementation – this SMART version of Rocchio is what we will refer to from now on.

Rocchio 1971 algorithm (SMART)

- Used in practice:

$$\begin{aligned}\vec{q}_m &= \alpha\vec{q}_0 + \beta\mu(D_r) - \gamma\mu(D_{nr}) \\ &= \alpha\vec{q}_0 + \beta\frac{1}{|D_r|}\sum_{\vec{d}_j\in D_r}\vec{d}_j - \gamma\frac{1}{|D_{nr}|}\sum_{\vec{d}_j\in D_{nr}}\vec{d}_j\end{aligned}$$

  q_m: modified query vector; q_0: original query vector; D_r and D_{nr}: sets of known relevant and non-relevant documents respectively; α, β, and γ: weights
- New query moves towards relevant documents and away from non-relevant documents.
- Tradeoff α vs. β/γ: If we have a lot of judged documents, we want a higher β/γ.
- Set negative term weights to 0.
- “Negative weight” for a term doesn’t make sense in the vector space model.

Positive vs. negative relevance feedback

- Positive feedback is more valuable than negative feedback.
- For example, set $\beta = 0.75$, $\gamma = 0.25$ to give higher weight to positive feedback.
- Many systems only allow positive feedback.

Relevance feedback: Assumptions

- When can relevance feedback enhance recall?
- Assumption A1: The user knows the terms in the collection well enough for an initial query.
- Assumption A2: Relevant documents contain similar terms (so I can “hop” from one relevant document to a different one when giving relevance feedback).

Violation of A1

- Assumption A1: The user knows the terms in the collection well enough for an initial query.
- Violation: Mismatch of searcher's vocabulary and collection vocabulary
- Example: cosmonaut / astronaut

Violation of A2

- Assumption A2: Relevant documents are similar.
- Example for violation: [contradictory government policies]
- Several unrelated "prototypes"
  - Subsidies for tobacco farmers vs. anti-smoking campaigns
  - Aid for developing countries vs. high tariffs on imports from developing countries
- Relevance feedback on tobacco docs will not help with finding docs on developing countries.

Relevance feedback: Assumptions

- When can relevance feedback enhance recall?
- Assumption A1: The user knows the terms in the collection well enough for an initial query.
- Assumption A2: Relevant documents contain similar terms (so I can "hop" from one relevant document to a different one when giving relevance feedback).

Relevance feedback: Evaluation

- Pick an evaluation measure, e.g., precision in top 10: $P@10$
- Compute $P@10$ for original query q_0
- Compute $P@10$ for modified relevance feedback query q_1
- In most cases: q_1 is spectacularly better than q_0!
- Is this a fair evaluation?

Relevance feedback: Evaluation

- Fair evaluation must be on "residual" collection: docs not yet judged by user.
- Studies have shown that relevance feedback is successful when evaluated this way.
- Empirically, one round of relevance feedback is often very useful. Two rounds are marginally useful.

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.

Exercise

- Do search engines use relevance feedback?
- Why?

Relevance feedback: Problems

- Relevance feedback is expensive.
  - Relevance feedback creates long modified queries.
  - Long queries are expensive to process.
- Users are reluctant to provide explicit feedback.
- It's often hard to understand why a particular document was retrieved after applying relevance feedback.
- The search engine Excite had full relevance feedback at one point, but abandoned it later.

Pseudo-relevance feedback

- Pseudo-relevance feedback automates the “manual” part of true relevance feedback.
- Pseudo-relevance feedback algorithm:
  - Retrieve a ranked list of hits for the user’s query
  - Assume that the top k documents are relevant.
  - Do relevance feedback (e.g., Rocchio)
- Works very well on average
- But can go horribly wrong for some queries.
  - Because of query drift
  - If you do several iterations of pseudo-relevance feedback, then you will get query drift for a large proportion of queries.

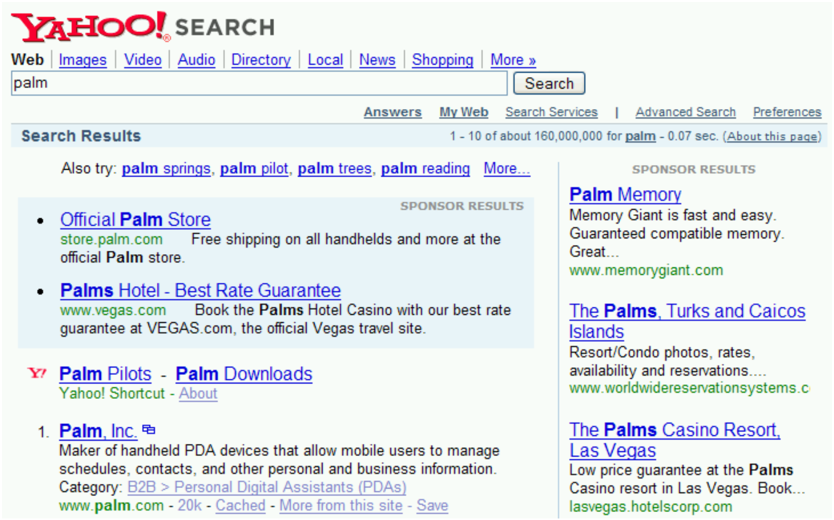
Pseudo-relevance feedback at TREC4

- Cornell SMART system
- Results show number of relevant documents out of top 100 for 50 queries (so total number of documents is 5000):

method	number of relevant documents
lnc.ltc	3210
lnc.ltc-PsRF	3634
Lnu.ltu	3709
Lnu.ltu-PsRF	4350

- Results contrast two length normalization schemes (L vs. l) and pseudo-relevance feedback (PsRF).
- The pseudo-relevance feedback method used added only 20 terms to the query. (Rocchio will add many more.)
- This demonstrates that pseudo-relevance feedback is effective on average.

Query expansion: Example


YAHOO! SEARCH
Web | Images | Video | Audio | Directory | Local | News | Shopping | More »
palm
Search
Answers My Web Search Services | Advanced Search Preferences

Search Results 1 - 10 of about 160,000,000 for palm - 0.07 sec. (About this page)

Also try: palm springs, palm pilot, palm trees, palm reading More...

SPONSOR RESULTS

- Official **Palm** Store
  store.palm.com Free shipping on all handhelds and more at the official **Palm** store.
- **Palms** Hotel - Best Rate Guarantee
  www.vegas.com Book the **Palms** Hotel Casino with our best rate guarantee at VEGAS.com, the official Vegas travel site.

Palm Pilots - **Palm** Downloads
Yahoo! Shortcut - About

1. **Palm**, Inc.
   Maker of handheld PDA devices that allow mobile users to manage schedules, contacts, and other personal and business information.
   Category: B2B > Personal Digital Assistants (PDAs)
   www.**palm**.com - 20k - Cached - More from this site - Save

SPONSOR RESULTS

Palm Memory
Memory Giant is fast and easy. Guaranteed compatible memory. Great...
www.memorygiant.com

The **Palms**, Turks and Caicos Islands
Resort/Condo photos, rates, availability and reservations....
www.worldwidereservationsystems.c

The **Palms** Casino Resort, Las Vegas
Low price guarantee at the **Palms** Casino resort in Las Vegas. Book...
lasvegas.hotelscorp.com

Types of user feedback

- User gives feedback on documents.
  - More common in relevance feedback
- User gives feedback on words or phrases.
  - More common in query expansion

Query expansion

- Query expansion is another method for increasing recall.
- We use “global query expansion” to refer to “global methods for query reformulation”.
- In global query expansion, the query is modified based on some global resource, i.e. a resource that is not query-dependent.
- Main information we use: (near-)synonymy

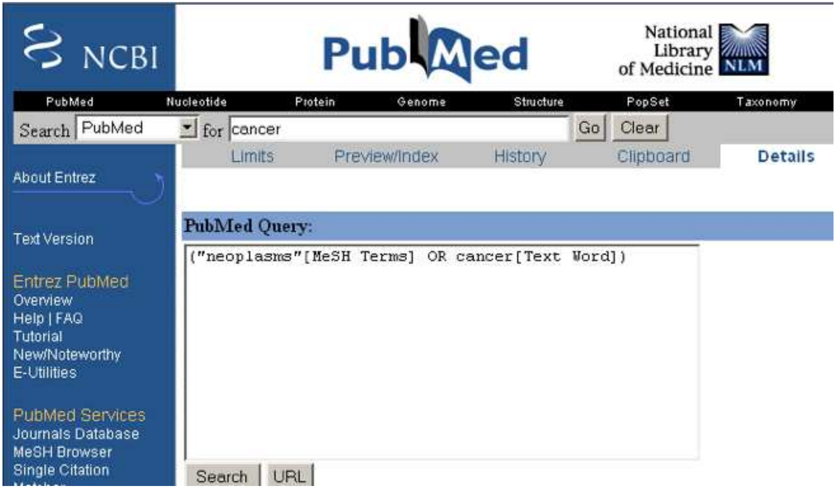
“Global” resources used for query expansion

- A publication or database that collects (near-)synonyms is called a thesaurus.
- Manual thesaurus (maintained by editors, e.g., PubMed)
- Automatically derived thesaurus (e.g., based on co-occurrence statistics)
- Query-equivalence based on query log mining (common on the web as in the “palm” example)

Thesaurus-based query expansion

- For each term t in the query, expand the query with words the thesaurus lists as semantically related with t.
- Example from earlier: HOSPITAL $\rightarrow$ MEDICAL
- Generally increases recall
- May significantly decrease precision, particularly with ambiguous terms
  - INTEREST RATE $\rightarrow$ INTEREST RATE FASCINATE
- Widely used in specialized search engines for science and engineering
- It's very expensive to create a manual thesaurus and to maintain it over time.

Example for manual thesaurus: PubMed


NCBI
PubMed
National Library of Medicine NLM
PubMed Nucleotide Protein Genome Structure PopSet Taxonomy
Search PubMed for cancer Go Clear
Limits Preview/Index History Clipboard Details
About Entrez
Text Version
Entrez PubMed
Overview
Help | FAQ
Tutorial
New/Noteworthy
E-Utilities
PubMed Services
Journals Database
MeSH Browser
Single Citation
Search URL

PubMed Query:

```
("neoplasms"[MeSH Terms] OR cancer[Text Word])
```

Automatic thesaurus generation

- Attempt to generate a thesaurus automatically by analyzing the distribution of words in documents
- Fundamental notion: similarity between two words
- Definition 1: Two words are similar if they co-occur with similar words.
    - “car” $\approx$ “motorcycle” because both occur with “road”, “gas” and “license”, so they must be similar.
- 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 is more robust, grammatical relations are more accurate.

Co-occurrence-based thesaurus: Examples

Word	Nearest neighbors
absolutely	absurd whatsoever totally exactly nothing
bottomed	dip copper drops topped slide trimmed
captivating	shimmer stunningly superbly plucky witty
doghouse	dog porch crawling beside downstairs
makeup	repellent lotion glossy sunscreen skin gel
mediating	reconciliation negotiate case conciliation
keeping	hoping bring wiping could some would
lithographs	drawings Picasso Dali sculptures Gauguin
pathogens	toxins bacteria organisms bacterial parasite
senses	grasp psyche truly clumsy naive innate

WordSpace demo on web

Soft cosine measure

- Use a matrix **S** that specifies the cosine similarity of basis vectors (i.e. of words) in Salton's vector space model.
- Definition 3: The similarity of two words is proportional to their cosine similarity.
    - "car" $\approx$ "motorcycle" iff cos("car", "motorcycle") ≈ 1.
- When the search engine supports non-orthogonal vector space model, then we can directly compute the soft cosine measure (SCM) between document vectors $\vec{u}$ and $\vec{v}$ by computing the matrix product $\vec{u}^{\mathsf{T}}\mathbf{S}\vec{v}$.
- Otherwise, we can expand the text query as follows:
    1. Translate the text query to a query vector $\vec{u}$.
    2. Compute $\vec{u'} = \vec{u}\mathbf{S}$.
    3. Translate $\vec{u'}$ back to a (now expanded) text query.
- Unlike a thesaurus based on word co-occurrences, the matrix **S** can be derived from word embeddings, the Levenshtein distance, and other measures of word similarity / relatedness.

SCM query expansion: Example

Query expansion using a Gramm matrix **S** that was built from the Google News word embeddings distributed with Word2Vec:

Original query : “**I did enact Julius Caesar: I was killed i’ the Capitol**”

Expanded query : “Give␣unto␣Caesar Brutus␣Cassius choreographers␣Bosco Julius␣Caesar therefore␣unto␣Caesar Marcus␣Antonius Caesarion Gallic␣Wars Marcus␣Crassus Antoninus Catiline Seleucus Gaius␣Julius␣Caesar Theodoric Marcus␣Tullius␣Cicero

⋮

Kenneth Philip Marcus Arthur Carl Fred Edward Jonathan Eric Frank Anthony William Richard Robert **enact Capitol killed** Ididn’t honestly myself **I I** my we **the** ’d ’m **did was**”

We can include only highly similar words in the expanded query. Search engines such as Apache Lucene make it possible to assign weights to words in text queries.

Query expansion at search engines

- Main source of query expansion at search engines: query logs
- Example 1: After issuing the query [herbs], users frequently search for [herbal remedies].
  - → “herbal remedies” is potential expansion of “herb”.
- Example 2: Users searching for [flower pix] frequently click on the URL photobucket.com/flower. Users searching for [flower clipart] frequently click on the same URL.
  - → “flower clipart” and “flower pix” are potential expansions of each other.

Take-away today

- Interactive relevance feedback: improve initial retrieval results by telling the IR system which docs are relevant / non-relevant
- Best known relevance feedback method: Rocchio feedback
- Query expansion: improve retrieval results by adding synonyms / related terms to the query
    - Sources for related terms: Manual thesauri, automatic thesauri, query logs

Resources

- Chapter 9 of IIR
- Resources at `https://www.fi.muni.cz/~sojka/PV211/` and `http://cislmu.org`, materials in MU IS and FI MU library
    - Daniel Tunkelang's articles on query understanding, namely on query relaxation and query expansion.
    - Salton and Buckley 1990 (original relevance feedback paper)
    - Spink, Jansen, Ozmultu 2000: Relevance feedback at Excite
    - Justin Bieber: related searches fail
    - Word Space
    - Schütze 1998: Automatic word sense discrimination (describes a simple method for automatic thesaurus generation)
    - Sidorov et al. 2014: Soft similarity and soft cosine measure: Similarity of features in vector space model
    - Charlet and Damnati 2017: SimBow at SemEval-2017 Task 3: Soft-Cosine Semantic Similarity between Questions for Community Question Answering (describes two matrices **S**)