Modern Information Retrieval

Relevance feedback and query expansion

Hamid Beigy

Sharif university of technology

March 7, 2025

Table of contents

1. Introduction
2. Relevance Feedback
3. The Rocchio algorithm
4. Evaluation of Relevance Feedback strategies
5. Local methods for query expansion
6. Global methods for query expansion
7. References

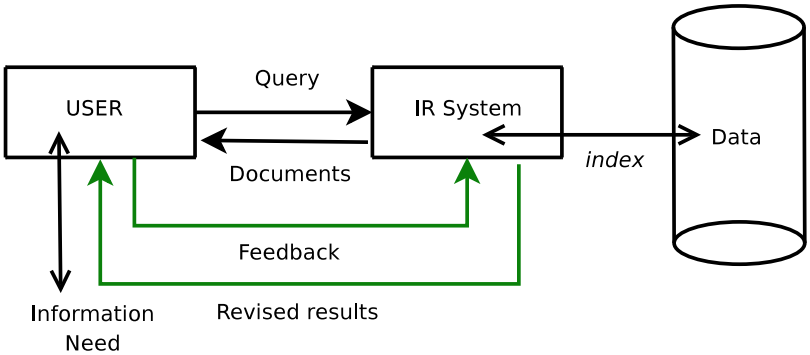
Introduction

Introduction

1. An information need may be expressed using different keywords (synonymy) such as aircraft vs airplane.
2. The same word can have different meanings (polysemy) such as Apple.
3. Vocabulary of searcher may not match that of the documents.
4. Solutions: refining queries manually or expanding queries automatically
5. Relevance feedback and query expansion aim to overcome the problem of synonymy.

Relevance Feedback

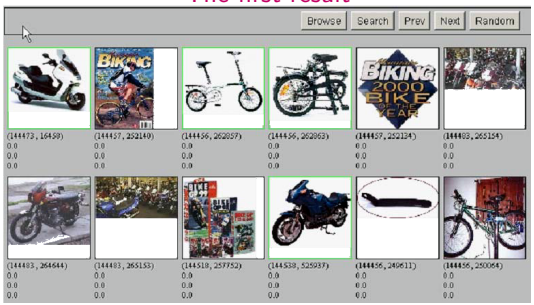
1. In relevance feedback, a set of document is given in response of a query.
2. Then the user specifies relevant and non-relevant documents.
3. The system refines the query and gives a new set of documents.



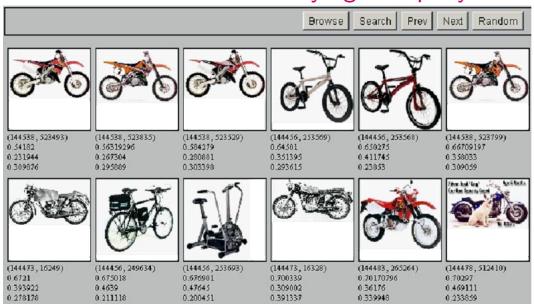
credit: Y. Parmentier

Relevance Feedback (example)

The first result



The result after modifying the query



Relevance feedback (example)

```
Query: New space satellite applications
+ 1. 0.539, 08/13/91, NASA Hasn't Scrapped Imaging
Spectrometer
+ 2. 0.533, 07/09/91, NASA Scratches Environment Gear From
Satellite Plan
3. 0.528, 04/04/90, Science Panel Backs NASA Satellite Plan,
But Urges Launches of Smaller Probes
4. 0.526, 09/09/91, A NASA Satellite Project Accomplishes
Incredible Feat: Staying Within Budget
5. 0.525, 07/24/90, Scientist Who Exposed Global Warming
Proposes Satellites for Climate Research
6. 0.524, 08/22/90, Report Provides Support for the Critics
Of Using Big Satellites to Study Climate
7. 0.516, 04/13/87, Arianespace Receives Satellite Launch
Pact From Telesat Canada
+ 8. 0.509, 12/02/87, Telecommunications Tale of Two
Companies
```

Relevance feedback (Term weights in query)

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

Relevance feedback (example)

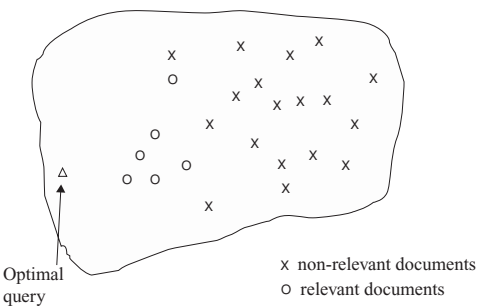
```
+ 1. 0.513, 07/09/91, NASA Scratches Environment Gear From
Satellite Plan
+ 2. 0.500, 08/13/91, NASA Hasn't Scrapped Imaging
Spectrometer
3. 0.493, 08/07/89, When the Pentagon Launches a Secret
Satellite, Space Sleuths Do Some Spy Work of Their Own
4. 0.493, 07/31/89, NASA Uses 'Warmâ ˜ Superconductors For
Fast Circuit
+ 5. 0.492, 12/02/87, Telecommunications Tale of Two
Companies
6. 0.491, 07/09/91, Soviets May Adapt Parts of SS-20 Missile
For Commercial Use
7. 0.490, 07/12/88, Gaping Gap: Pentagon Lags in Race To
Match the Soviets In Rocket Launchers
8. 0.490, 06/14/90, Rescue of Satellite By Space Agency To
Cost \$90 Million
```

The Rocchio algorithm

The Rocchio algorithm

1. This algorithm is a standard algorithm for relevance feedback proposed by Salton in 1970
2. This algorithm integrates a measure of relevance feedback into vector space model
3. The idea is to find a query vector q_{opt} by
   - maximizing the similarity with relevant documents and
   - minimizing the similarity with non-relevant documents.
4. This can be obtained via $q_{opt} = argmax_q \quad [sim(q, C_r) - sim(q, C_{nr})]$
5. By using cosine similarity, we obtain

$$q_{opt} = \frac{1}{|C_r|} \sum_{d_j \in C_r} d_j - \frac{1}{|C_{nr}|} \sum_{d_j \in C_{nr}} d_j$$



1. The problem is that the set of relevant documents is unknown
2. Instead, we can produce the modified query m:

$$q_m = \alpha q_0 + \beta \frac{1}{|D_r|} \sum_{d_j \in D_r} d_j - \gamma \frac{1}{|D_{nr}|} \sum_{d_j \in D_{nr}} d_j$$

where

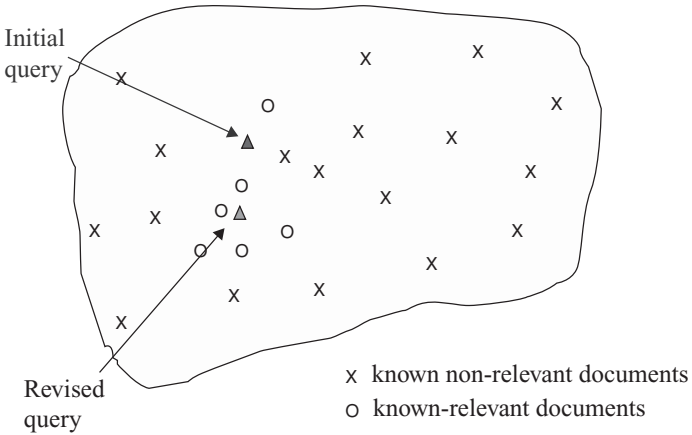
- q_0 : the original query vector
- D_r : the set of known relevant documents
- D_{nr} : the set of known non-relevant documents
- α, β, γ are balancing weights

The Rocchio algorithm

1. In Rocchio algorithm, negative weights are usually ignored ($\gamma = 0$)
2. This relevance feedback improves both recall and precision
3. In order to reach high recall value, many iterations are needed
4. These weights are determined empirically and usually set as

   $\alpha = 1 \quad \beta = 0.75 \quad \gamma = 0.15$

5. Positive feedback is usually more valuable than negative feedback: $\beta > \gamma$



1. Alternative to the Rocchio algorithm, use a document classification instead of a vector space

$$P(x_t = 1|R = 1) = \frac{|VR_t|}{|VR|}$$

$$P(x_t = 0|R = 0) = \frac{n_t - |VR_t|}{N - |VR|}$$

where

- $P(x_t = 1)$ shows the probability of a term t appearing in a document
- $R = 1$ shows that the document is relevant
- $R = 0$ shows that the document is non-relevant
- N is the total number of documents
- n_t is the number of documents containing term t
- VR is set of known relevant documents
- VR_t is set of known relevant documents containing term t

When to use Relevance Feedback

1. Relevance Feedback does not work when
   - the query is misspelled
   - we want cross-language retrieval
   - the vocabulary is ambiguous
2. This implies that users do not have sufficient initial knowledge

Relevance Feedback and the web

1. A few web IR systems use relevance feedback because
   - hard to explain to users
   - users are mainly interested in fast retrieval
   - users usually are not interested in high recall
2. Now, they are using an implicit feedback such as clickstream-based feedback

Evaluation of Relevance Feedback strategies

1. Evaluation strategies for relevance feedback
   - Comparative evaluation

     comparing prec/recall graph after processing q_0 and q_m

     This usually increases +50% of mean average precision
   - Residual collection (the set of documents minus those assessed relevant)

     Fair evaluation must be on residual collection: docs not yet judged by user.
   - Using two similar collections

     The first collection is used for querying and giving relevance feedback and the second collection is used for comparative evaluation
   - User studies

     time-based comparison of retrieval for measuring user satisfaction

Local methods for query expansion

Pseudo Relevance Feedback (blind relevance feedback)

1. There is no need of an extended interaction between the user and the system
2. Pseudo-relevance feedback automates the manual part of true relevance feedback.
3. We can
   - 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)

Indirect Relevance Feedback

1. Uses evidences rather than explicit feedback such as the number of clicks on a given retrieved document
2. Not user-specific
3. More suitable for web IR, since it does not need an extra action from the user.
   - Clicks on links are assumed to indicate that the page is more likely to be relevant
   - Click-rates can be gathered globally for clickstream mining

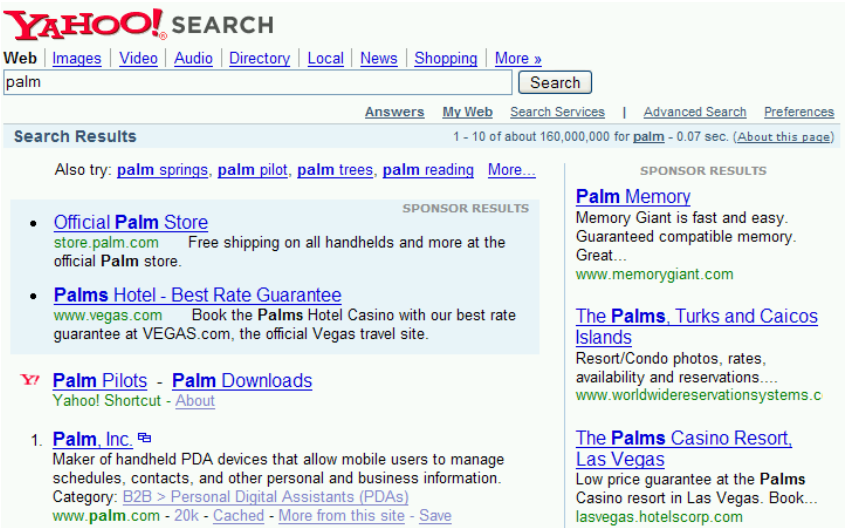
Global methods for query expansion

Vocabulary tools for query reformulation

Tools displaying:

1. a list of close terms belonging to the dictionary
2. information about the query words that were omitted (stop-list)
3. the results of stemming
4. this approximating debugging environment

Query expansion



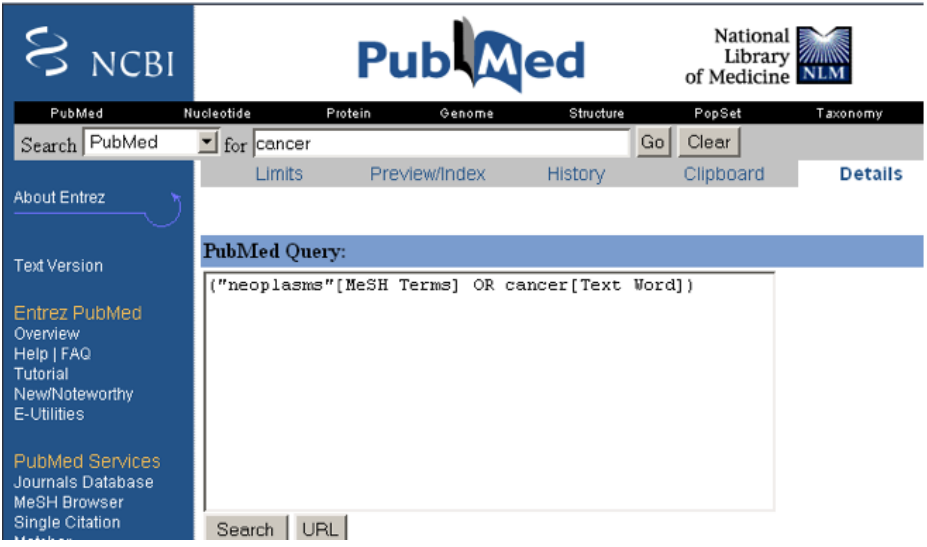
Query logs and thesaurus

1. Users select among query suggestions that are built either from **query logs** or **thesaurus**
2. Replacement words are extracted from thesaurus according to their proximity to the initial query word
3. Thesaurus can be developed
   - manually
   - automatically

Manual thesaurus

1. Maintained by publishers (e.g. PubMed)
2. Widely used in specialized search engines for science and engineering
3. It's very expensive to create a manual thesaurus and maintain it over time
4. Roughly equivalent to annotation with a controlled vocabulary.

Manual thesaurus



Automatic thesaurus generation

1. Analyze of the collection for building the thesaurus automatically
   - Using word co-occurrences (co-occurring words are more likely to belong to the same query field)
   - Using a shallow grammatical analyzes to find out relations between words
2. Co-occurrence-based thesaurus are more robust, but grammatical-analyzes thesaurus are more accurate

Building a co-occurrence-based thesaurus

1. We build a term-document matrix A where $A[t, d] = w_{t,d}$ (*e.g.* normalized $tf - idf$)
2. We then calculate $C = A.A^T$

$$C = \begin{pmatrix} c_{11} & \cdots & c_{1n} \\ \vdots & \ddots & \vdots \\ c_{m1} & \cdots & c_{mn} \end{pmatrix}$$

c_{ij} is the similarity score between terms i and j

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

References

Reading

1. Chapters 9 of Information Retrieval Book[1].
2. Section 7.1 of Search Engines - Information Retrieval in Practice Book[2].

[1] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze (2008). *Introduction to Information Retrieval*. New York, NY, USA: Cambridge University Press.

[2] W. Bruce Croft, Donald Metzler, and Trevor Strohman (2009). *Search Engines - Information Retrieval in Practice*. Pearson Education.

References

- Croft, W. Bruce, Donald Metzler, and Trevor Strohman (2009). *Search Engines - Information Retrieval in Practice*. Pearson Education.
- Manning, Christopher D., Prabhakar Raghavan, and Hinrich Schütze (2008). *Introduction to Information Retrieval*. New York, NY, USA: Cambridge University Press.

Questions?