Query Expansion Techniques

(Relevance Feedback, Thesaurus, Semantic Network)

(COSC 488)

Nazli Goharian

nazli@ir.cs.georgetown.edu

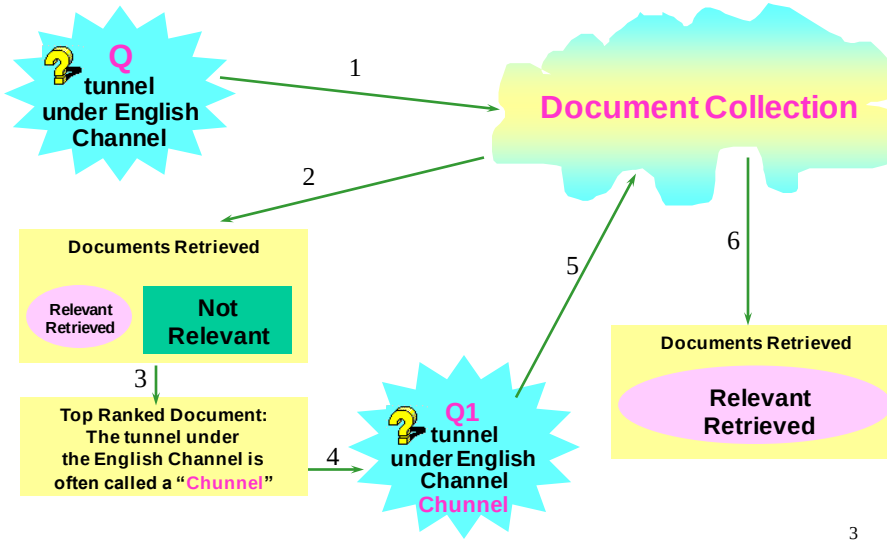
1

Relevance Feedback

- The modification of the search process to improve the effectiveness of an IR system
- Incorporates information obtained from prior relevance judgments
- Basic idea is to do an initial query, get feedback from the user (or automatically) as to what documents are relevant and then add term from known relevant document(s) to the query.

2

Blind Relevance Feedback Example



3

Feedback Mechanisms

- Automatic (Pseudo/ Blind)
  - The "good" terms from the "good", top ranked documents, are selected by the system and added to the users query.
- Semi-automatic
  - User provides feedback as to which documents are relevant (via clicked document or selecting a set of documents); the "good" terms from those documents are added to the query.
  - Similarly terms can be shown to the user to pick from.
  - Suggesting new queries to the user based on:
    - Query log
    - Clicked document (generally limited to one document)

4

Pseudo Relevance Feedback Algorithm

- Identify "good" (*N* top-ranked) documents.
- Identify all terms from the N top-ranked documents.
- Select the "good" (*T* top) feedback terms.
- Merge the feedback terms with the original query.
- Identify the top-ranked documents for the modified queries through relevance ranking.

5

Sort Criteria

- Methods to select the "good" terms:
  - n*idf (a reasonable measure)
  - f*idf
  - …..

  where:
  - *n: is* number of documents in relevant set having term *t*
  - *f:* is frequency of term t in relevant set

6

Example

- Top 3 documents
  - d1: A, B, B, C, D
  - d2: C, D, E, E, A, A
  - d3: A, A, A
  - Assume *idf* of A, B, C is 1 and D, E is 2.

Term	n	f	n*idf	f*idf
A	3	6	3	6
B	1	2	1	2
C	2	2	2	2
D	2	2	4	4
E	1	2	2	4

based on n*idf:

Top 2 terms
D
A

Top 3 terms:
D
A
C or E

7

Original Rocchio Vector Space Relevance Feedback [1965]

- Step 1: Run the query.
- Step 2: Show the user the results.
- Step 3: Based on the user feedback:
  - add new terms to query or increase the query term weights.
  - Remove terms or decrease the term weights.

- ***Objective*** => *increase the query accuracy.*

8

Rocchio Vector Space Relevance Feedback

$$Q' = \alpha Q + \beta \sum_{i+1}^{n_1} R_i - \gamma \sum_{i=1}^{n_2} S_i$$

- Q: original query vector
- R: set of relevant document vectors
- S: set of non-relevant document vectors
- α, β, γ: constants (Rocchio weights)
- Q': new query vector

9

Variations in Vector Model

$$Q' = \alpha Q + \beta \sum_{i+1}^{n_1} R_i - \gamma \sum_{i=1}^{n_2} S_i$$

Options:

$\alpha = 1, \beta = \frac{1}{|R|}, \gamma = \frac{1}{|S|}$

$\alpha = \beta = \gamma = 1$

- Use only first *n* documents from R and S
- Use only first document of S
- Do not use S ($\gamma = 0$)

10

Implementing Relevance Feedback

- First obtain top documents, do this with the usual inverted index
- Now we need the top terms from the top *X* documents.
- Two choices
  - Retrieve the top *x* documents and scan them in memory for the top terms.
  - Use a separate doc-term structure that contains for each document, the terms that will contain that document.

11

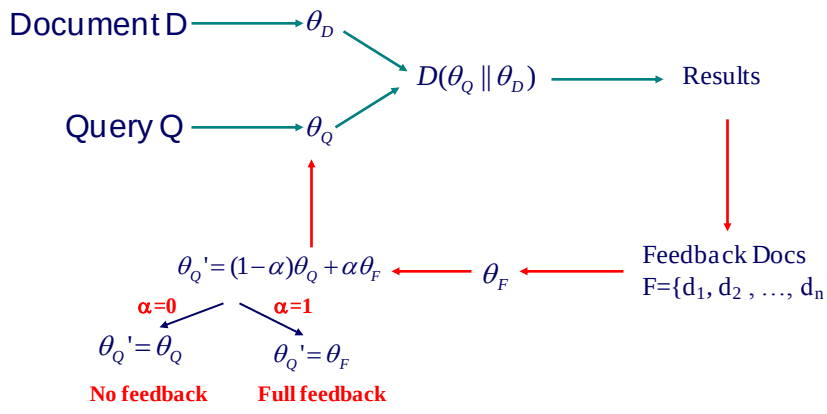
Relevance Feedback in Probabilistic Model

- Need training data for R and r (unlikely)
- Some other strategy like VSM can be used for the initial pass to get the top *n* docs, as the relevant docs
  - *R* can be estimated as the total relevant docs found in top n
  - *r* is then estimated based on these documents
- Query can be expanded using the *expanded Probabilistic Model* term weighting
- Options: re-weighting initial query terms; adding new terms w/wo re-weigthing initial query terms

12

Pseudo Relevance Feedback in Language Model

(from: Manning based on Viktor Lavrenko and Chengxiang Zhai)

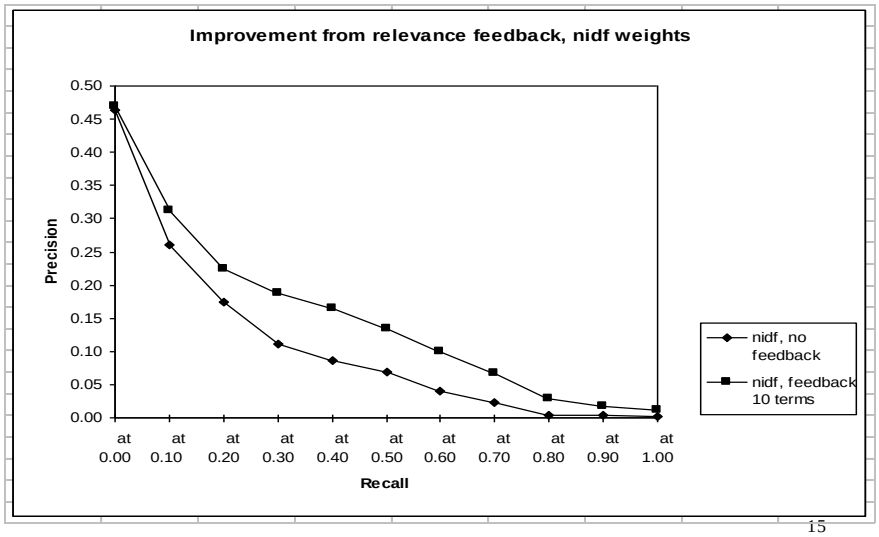


Relevance Feedback Modifications

- Various techniques can be used to improve the relevance feedback process.
  - Number of Top-Ranked Documents
  - Number of Feedback Terms
  - Feedback Term Selection Techniques
  - Iterations
  - Term Weighting
  - Phrase versus single term
  - Document Clustering
  - Relevance Feedback Thresholding
  - Term Frequency Cutoff Points
  - Query Expansion Using a Thesaurus

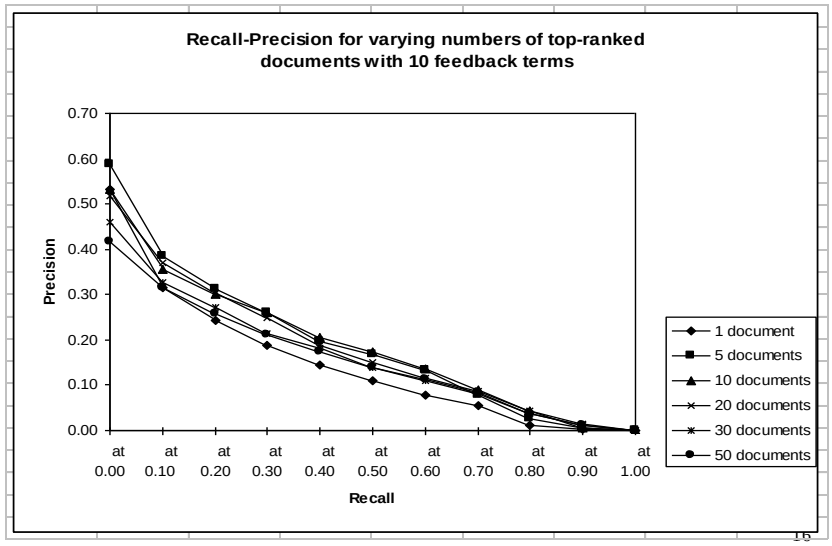
14

Relevance Feedback Justification



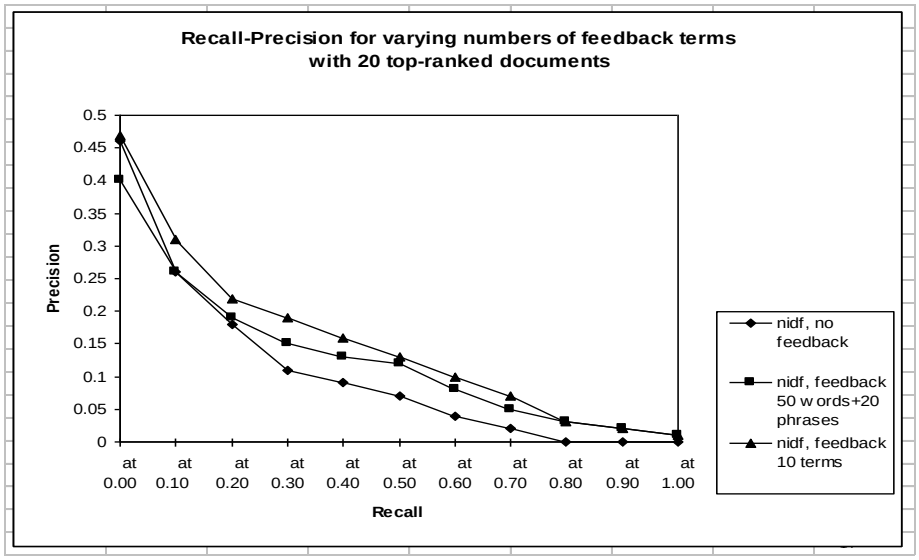
15

Number of Top-Ranked Documents



15

Number of Feedback Terms



Summary of Relevance Feedback

- Pro
  - Relevance feedback usually improves average precision by increasing the number of good terms in the query (generally 10-15% improvement in traditional IR search)
- Con
  - More computational work
  - Easy to decrease Precision (one horrible word can undo the good caused by lots of good words).

18

Thesauri

- It is intuitive to use thesauri to *expand* a query to enhance the accuracy.
- A query about "dogs" might well be expanded to include "canine" if a thesauri was consulted.
- Problem: easily a "bad" word can be added. A synonym for "dog" might well be "pet" and then the query would be too generic.

19

Thesauri

- Available Machine readable
  - Use a readily available machine-readable form of a thesauri (e.g. Roget's, etc.).
- Custom made
  - build a thesaurus automatically in a language independent fashion

20

Thesaurus Generation with Term Co-occurrence

- Thesaurus is generated by finding similar terms.
- terms that *co-occur* with each other over a threshold are considered *similar*.
- Term-Term similarity matrix is created, having SC between every term t_i with t_j

Term Vectors (term-doc mapping):

t_1 <1 1>

t_2 <0 1>

SC (t_1, t_2) = <0 1>.<1 1> = 1 *dot product*

21

Expanding Query using Term Co-occurrence

- For a given term t_i, the top *t* similar terms are picked.
- These words can now be used for query expansion.

- Problems:
  - A very frequent term will co-occur with everything
  - Very general terms will co-occur with other general terms

22

Semantic Networks

- Attempts to resolve the *mismatch* problem
- Instead of matching query terms and document terms, measures the *semantic distance*
- Premise: Terms that share the same meaning are closer (smaller distance) to each other in semantic network

See publicly available tool, WordNet (*www.cogsci.princeton.edu/~wn*)

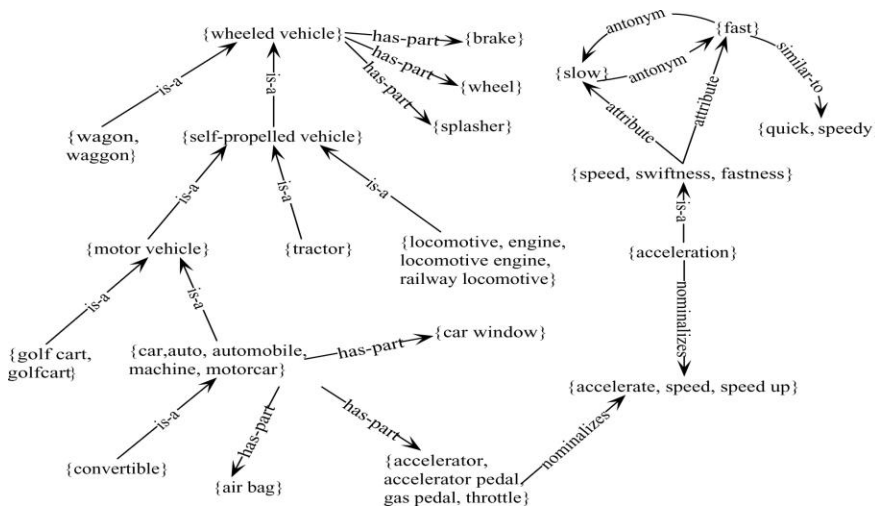
23

Semantic Networks

- Builds a network that for each word shows its relationships to other words (may be phrases).
- For *dog* and *canine* a *synonym* arc would exist.
- To expand a query, find the word in the semantic network and follow the various arcs to other related words.
- Different *distance measures* can be used to compute the distance from one word in the network to another.

24

WordNet



based on Word Sense Disambiguation Survey by R. Navigli, ACM Computing Surveys, 2009

Types of Links in Wordnet

- Synonyms
  - dog, canine
- Antonyms (opposite)
  - night, day
- Hyponyms (is-a)
  - dog, mammal
- Meronyms (part-of)
  - roof, house
- Entailment (one entails the other)
  - buy, pay
- Troponyms (two words related by entailment must occur at the same time)
  - limp, walk

26

Query Expansion using Concepts & External Sources

- L. Jia, C. T. Yu, and W. Zhang, UIC at TREC 2008 blog track.
- W. Zhang and C. Yu, UIC at TREC 2007 blog track.

Adding synonyms of the concepts identified in query.

- Find a Wikipedia entry page for each query concept. Then add to initial query:
  - *Title* of Wikipedia page
  - Terms appearing frequently in around the original query terms in Wikipedia entry page, Google search results, blog posts.

27

Query Expansion using Concepts & External Sources

- Feedback terms from 10 documents from an external resource (Wikipedia, news resource aligned with of blog posts.

V. Lavrenko and W. B. Croft, "Relevance based language models," in Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2001), pp. 120–127, 2001.

Expanded query model θq obtained by combining expanded and original models

$$P(t|\theta q) = \lambda P(t|\theta q^e) + (1 - \lambda) P(t|\theta q^o)$$

λ controls the mixture between the two models

"the top ad hoc search performances in the TREC 2007 and 2008 Blog tracks"

28

Query Expansion using Machine Learning

Q. Zhang, B.Wang, L.Wu, and X. Huang, Fdu at trec 2007: opinion retrieval of blog track.

Top 120 posts; top 400 terms → term vector

Term vector features: term and document frequency info

→ a set of 200 expansion terms was selected using a SVM (Support Vector Machine) classifier

29

Query Expansion in Microblog Search

- Twitter average query length 1.64 → Query expansion techniques can improve understanding user query intent

Various approaches proposed, using:

- *Term statistics, such as TF*
- *Temporal feature*
- *External sources, such as Wikipedia, News,….*
- *…….*

31

Relevance Feed in Blog Search

Y. Lee, S.-H. Na, and J.-H. Lee, "An improved feedback approach using relevant local posts for blog feed retrieval," in Proceeding of the ACM conference on Information and Knowledge Management (CIKM 2009), pp. 1971–1974, 2009.

Problems:

topic bias incurred by expanding terms from highly ranked blog posts

topic drift incurred by expanding terms from all posts of each blog

Solution: Diversity oriented query expansion

Use top m retrieved posts from the top k retrieved blogs as the pseudo-relevance feedback set

32

Query Expansion using Passages

- Using retrieved passages for feedback (in Blog Search)
  - *Y. Lee, S.-H. Na, J. Kim, S.-H. Nam, H.-Y. Jung, and J.-H. Lee, KLE at TREC 2008 Blog track: blog post and feed retrieval.*
- Chose highest scoring passages in posts, augmented with a fixed-length left and right context.
  - *S.-H. Na, I.-S. Kang, Y. Lee, and J.-H. Lee, "Applying completely-arbitrary passage for pseudo-relevance feedback in language modeling approach," in Proceedings of the Asia Infomation Retrieval Symposium, pp. 626–631, 2008.*

33

Summary

- Query expansion techniques, such as relevance feedback, Thesauri, WordNet (Semantic Network) can be used to find "hopefully" good words for users
- They are mostly effective on short and non-specific queries
- Using user intervention for the feedback improves the results

34