INFSCI 2140
Information Storage and Retrieval
Lecture 8: Improving Search Effectiveness

Peter Brusilovsky
http://www2.sis.pitt.edu/~peterb/2140-061/

Ad-hoc IR in text-oriented DS

- The context (L1)
- Querying and matching (L2,L3)
- How it all works internally (L4,L10)
- How to evaluate results (L5)
- Better organization and visualization of search results (L7)
- Improving search results: UI, RF, QE (L8); UM (L9)
Overview

- Iterative search
- Interfaces to support query formulation and search process
- Relevance Feedback
- Query Expansion

How to improve the performance (precision, recall) of IR systems?

- The only thing ad-hoc IR can do is to return results for a query…
- What if a query returned poor results?
  - Help the user to produce better query (UI)
  - Produce better query for the user (QE)
  - Work with user on improving the query (RF)
- Who will do it?
  - User-system interaction
  - System alone (system-system interaction)
Iterative Search, Relevance Feedback, Query Expansion

Step diagram for traditional information access process
Iterative Search

- Search results can be improved by producing better and better queries
- How to get the user to produce better queries?
  - Educate the user about iterative search and query improvement
  - Provide better interface for query formulation
  - Provide an interface that supports iterative search

Query formulation

- **Boolean system:**
  - Identify concepts; expand concepts with synonyms; add logical operators
- **Ranked output system:**
  - Create a natural language or narrative statement of the information need (often very brief, especially in web environment)

Adopted from E. Rasmussen
User Interaction in a Boolean System

- Enter search statement
- View results (postings for terms and total)
- Available strategies
  - Modify strategy (topical)
    - Broader---add synonyms
    - Narrower---add concept, replace terms with more specific ones
  - Modify strategy (scope)
    - Specify journals, year, language, subject code
  - View sample citations (pearl growing)

Adopted from E. Rasmussen

User Interaction in a Ranked Output System

- Enter search statement
- View ranked list
- Strategies
  - Add or delete terms from search statement
  - Relevance feedback (if available)

Adopted from E. Rasmussen
Improving Search Interfaces

- Classic ad-hoc search is oriented to old “teletype” interface style
  - Query is typed in
  - Results are returned as a flow of text
- Interfaces have improved over years
- Can we improve search interfaces too?

Form-Based Query Formulation

- AND
- OR
- NOT
Query Previews

- The choice of right word or combination may be hard - it is not known in advance whether it will result in too many or too few results
- Quick query previews can help to choose the right set of query terms
- Essential if the search could be slow (UMD movie)

Venn Diagrams for QF

- Venn diagrams were proposed to improve Boolean query formulation
Two or more filters in sequence create the semantics of a conjunct (AND).
Two or more in parallel create a disjunct (OR).

Direct manipulation for iterative search?

- How we can use direct manipulation in the classic ad-hoc search process?

- The case of Stanford Digital Library (CHI'97)
System-Driven Query Improvement

- Relevance Feedback
- Query Expansion
- May be based on
  - Feedback information from the user
  - Information derived from documents
    - Local (documents initially retrieved)
    - Global (document collection as a whole)
  - Information derived from world knowledge

Adopted from E. Rasmussen

Relevance Feedback and Query Expansion - 3 points of view

- Pragmatic view
  - Modern Information Retrieval, Baesa-Yates
- Designer’s view
  - Bob Korfhage
- Interaction view
  - Nick Belkin
Korfhage’s view

- QE/RF is a manipulation with query to improve search results
- Main source of information - user
- What can be changed – query, document, algorithms
- What kind of manipulation – re-weighting, adding/removing, altering
- User profiles and genetic algorithms

Belkin’s view

- Information retrieval is an interaction between a human and information [system]
- Query is simply the first step in a dialogue - a part of user model that the system can build
- More interaction is required to update models of dialog partners
Pragmatic Point of View

- QE/RF are technique of improving query to achieve better results (precision or recall)
- The idea is to “steer” the query closer to the subspace of the relevant documents
- How to steer?
  - Boolean operation (and, or)
  - Vector operations:
    - Project (remove), Add, Re-weight

Query Expansion

- Query expansion (term expansion) is the process of supplementing the original query with additional terms
- The idea: add some extra “good” terms to a query in a hope that it will bring more results or better precision
- Possible sources of good terms
  - Automatic: Properties of the document space and term distribution
  - User-based: Manual, Relevance feedback
Manual Query Expansion

- Boolean search strategies
- Building blocks
  - Identify Concepts
  - Decompose concepts
  - Join with OR, AND
- Citation pearl growing
  - Direct/specific search term
  - Find a citation; review, augment, and iterate
  - \((\text{term}\_\text{facet})A \text{ AND } (\text{term}\_\text{facet})B \text{ AND } (\text{term}\_\text{facet})C\)
Automatic Query Expansion

- Local analysis: documents and term distribution in the current search
  - Local Clustering
  - Local Context Analysis
- Global analysis: document and term distribution in the whole space
  - Collection-dependent
  - Collection-independent

QE with local clustering

- Idea: add terms that are similar to good terms in the context of good documents
- Step 1: Cluster all terms using similarity metrics based of co-occurrence in documents
- Step 2: For each term in a query add M nearest neighbors in the cluster to a query
QE with local context analysis

- Idea - add concepts “similar” to the whole query
- Step 1: Get N top ranked passages using original query
  - Document is divided into small chunks
- Step 2: Calculate similarity between each concept from the passage and the whole query using a version of TF*IDF
- Step 3: Add top concepts to the query

QE with Collection-Dependent Analysis (Statistical)

- Similar ideas based on global analysis of terms in document collection
- Simple clustering and term co-occurrence produce poor results
- Use similarity thesaurus (terms clustered as documents with inverse indexing)
- Represent a query in the space of concepts and find terms that are most close in this space to the whole query
QE with Collection-Independent Analysis

- Use Domain Thesaurus
  - Domain Thesaurus represents relationship between terms
- Modern examples
  - MeSH, INSPEC, WordNet, Roget's
- Use Natural Language Processing
  - Find Named Entities (person, organization, place, time)
  - Noun Phrases

Interactive Query Expansion

- Based on search results
  - Extracting terms from retrieved records based on the term frequency.
- Collection Dependent
  - User vs. System
    - User, thesaurus expansion in dialog (E command)
    - Systems, recommending systems (candidate terms)
- Collection Independent
  - Expert systems
    - CANSEARCH, CONIT, TOMESEARCHER, etc.
  - Thesauri

Adapted from E. Rasmussen
Relevance Feedback: The Idea

- The user indicates the relevance of a set of documents providing *relevance judgement (feedback)*
- The IR system learns something about the user preferences and modifies its retrieval behaviors
  - May be done with or without expansion of the query (adding new terms)
- Then a new set of documents is presented and the retrieval process starts again

Kinds of Relevance Feedback

- Positive feedback
  - Mark relevant documents
- Negative feedback
  - Mark irrelevant documents
- Mixed feedback
  - Positive and negative
  - Rating on some scale (cold/hot/lukewarm)
Using Relevance Feedback

- The system can use positive relevance judgment trying to obtain more documents similar to those judged relevant.
- The system can use negative relevance judgments trying to avoid documents similar to the one that were rejected.

Positive of Negative?

- **Positive feedback (+ve)**
  - Shields users from query reformulation task
  - Breaks searching task into series of steps
  - Emphasizes relevant terms and de-emphasizes non-relevant ones
  - Can give significant performance improvement

- **Negative feedback (-ve)**
  - Takes control away from the searcher
  - Can move search in unanticipated directions and degrade performance
Relevance Feedback: Where?

- Where can we apply the information provided by the user?
- Query
- Profile
- Document representation
- Retrieval algorithm

Modifying the Query

- This is what we can also call user-based query expansion
- It is the simplest way
- It has no lasting impact on the system (that is a mixed blessing)
- Explored by Salton and Rocchio
  - Rocchio Algorithm
Modifying the user profile

- Profile - a long term representation of user interests
  - We will learn details later
- These modifications last
- User profile and query often have the same or nearly the same representation
  - it is possible to use the same techniques
- The modifications should not be made on the basis of a single query

Modifying the document representation

- Modifications that last and can effect the behavior of the system for all the users
- It can be accepted if the community of users is a closed community of experts
- Methods are similar to query modification
  - Some variants of Rocchio algorithm can be used
Modifying the search algorithm

- Possible to change
  - Algorithms parameters (easy to undo)
  - The algorithm itself (this modifies deeply the behavior of the system)
- It is something to do very carefully
  - Rarely used

RF for Query Modification Types

- Reweight original query terms
- Reweight original terms, add new terms (with or without limit, deletions)
- Drop original query terms, use only terms from retrieved set
  - ‘query by example’ uses a document as a query
  - ‘more like this’
Rocchio Algorithm (IR)

\[ Q' = Q + \frac{1}{n_1} \alpha \sum_{i=1}^{n_1} R_i - \frac{1}{n_2} \beta \sum_{i=1}^{n_2} S_i \]

where

- \( Q \) is the vector of the initial query
- \( R_i \) is the vector for relevant document
- \( S_i \) is the vector for the irrelevant documents
- \( \alpha, \beta \) are Rocchio’s weights

Relevance Feedback: Vector View

- \( D_r \) - set of relevant documents \( \{d_r\} \)
- \( D_n \) - set of non-relevant documents \( \{d_n\} \)
- Rocchio’s Formula:
  \[ q_m = \alpha q + \left( \frac{\beta}{|D_r|} \right) \sum d_r - \left( \frac{\gamma}{|D_n|} \right) \sum d_n \]
- Ide’s Regular Formula
  \[ q_m = \alpha q + \beta \sum d_r - \gamma \sum d_n \]
- Ide’s Formula
  \[ q_m = \alpha q + \beta \sum d_r - \gamma \sum \max_{\text{non-rel}} (d_j) \]
Problems - User Side

- **Rating**
  - More information vs. user overload
- **Supporting iterative search**
  - The user gets tired after 3 or 4 iterations
  - The user prefers to have a sort of “incremental interface” with the new documents highlighted in order to avoid having to scan the same documents again and again.

Interface for RF Search

- What can help?
- Better interface
- The use of long-term profiles

New documents

Documents already judged relevant
Pseudo-Relevance Feedback

- Relevance feedback mechanisms used without real user involvement
- System simply assumes that its top-ranked documents are relevant, and uses these documents to augment the query with a relevance feedback ranking algorithm
- Found to be highly effective in some settings, especially if the original query statement is long and precise

Implicit Relevance Feedback

- Users do not like to provide explicit relevance feedback
  - Considered as side activity, not related to the core task
- The system can estimate document relevance from user actions
  - Reading time, more information, printing, bookmarking
- More about that in user modeling lecture

Adopted from E. Rasmussen
Group Relevance Feedback

- Using relevance judgments from a large number of different users to rate or rank information of general interest
- Variation: use only similarity among relevance judgments by people with similar tastes, ignoring the representation of the information itself
- Highly effective for rating information in which taste plays a major role, such as movie and music recommendations