Part 8: Relevance Feedback

Francesco Ricci

Most of these slides comes from the course:

Information Retrieval and Web Search,
Christopher Manning and Prabhakar Raghavan
Content

- Local methods
  - Relevance feedback
  - Pseudo relevance feedback
  - Indirect (implicit) relevance feedback

- Global methods
  - Query expansion
    - Thesauri
    - Automatic thesaurus generation
    - Query log mining
Relevance Feedback

- Relevance feedback: *user feedback on relevance of docs in initial set of results*
  - User issues a (short, simple) query
  - The user marks some results as relevant or non-relevant
  - The system *computes a better query representation of the information need based on feedback*
  - Relevance feedback can go through one or more iterations

- **Idea:** it may be difficult to formulate a good query when you don’t know the collection well, so iterate.
Example: search images

Google images

Images   Show options...

Bikes Photos
AlthEye.com Photos of Bikes in cities all over the world

Related searches: cartoon bike, bmx bike, mountain bike, bicycle

Feedback

Available bikes and 1207 x 753 - 326k - gif www.fanweb.com
Find similar images

Six Crazy Concept 640 x 483 - 67k - jpg wired.com
Find similar images

sneaker meets bike 500 x 375 - 52k - jpg enveapparel...
Find similar images

The Youth Clinic 379 x 385 - 152k - png suitcaseclinic.org
Find similar images

Mountain bikes 406 x 380 - 31k - jpg adventure...
Find similar images
New Interface
Preference-based: DieToRecs

There are two ways to gain easily travel recommendations:
- Follow one of the alternatives you are interested and you will receive detailed offers.
- Rate the alternatives, click "Submit" and you will receive additional alternatives.

[Destinations]:
1. Ischgl, Tirol
   (Accommodation): 4***
   (Min - Max [Euro]): 0.0 - 156.0
   I like this!
2. Ottenbruck
   (Accommodation): 4***
   (Min - Max [Euro]): 0.0 - 0.0
   I like this!
3. Innsbruck
   (Accommodation): 4***
   (Min - Max [Euro]): 0.0 - 156.0
   I like this!
4. Längenfeld
   (Accommodation): 3***
   (Min - Max [Euro]): 0.0 - 0.0
   I like this!
5. Kastental
   (Accommodation): 4***
   (Min - Max [Euro]): 0.0 - 0.0
   I like this!
6. Tux
   (Accommodation): 4***
   (Min - Max [Euro]): 0.0 - 0.0
   I like this!

[Suggest more like this!]
Submit your rates to improve our suggestions.

[Next]

[Go Back]

History of your inspiration visits:
1. Kemmerich, 2009
2. Trip to Salzburg
3. Trip to Salzburg
4. 10.06.2009 10.21.45
5. 10.06.2009 10.21.45

Search for travel items:

[Ricci et al., 2006]
Exploratory Search: Example
Critiquing

Entree Results

We recommend:

Dave's Italian Kitchen (map)
906 Church St. (bet. Ridge & Sherman Aves.), Evanston, 708-864-6000

Italian

below $15

Fair Decor, Excellent Service, Excellent Food, No Reservations, Weekend Brunch, Carry in Wine and Beer, Wheelchair Access, Long Drive

less $15

more

creative

traditional

quieter
cuisine

For other suggestions, select:

Dave's Italian Kitchen
Gusto Italiano
Carlucci
Spavone's Seven Hills

Dancing Noodles Cafe
La Sorella di Francesca
Village

Anna Maria Pasteria
Mia Francesca
Rosebud

Salvatore's
Critiquing Interaction

- Initial preferences
- System *shows* K examples
- User picks the final choice
- User revises the preference model by critiquing examples

[Pu et al., 2006]
Key concept: 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.
- Definition: Centroid

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

where \( C \) is a set of documents.
The centroid is not normalized
The Theoretically Best Query

x non-relevant documents
O relevant documents
Rocchio Algorithm

- The Rocchio algorithm uses the vector space model to pick a relevance feedback query.
- Rocchio seeks the query $\vec{q}_{opt}$ 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 docs marked relevant $C_r$ and non-relevant $C_{nr}$ – the solution is:

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

- Problem: we don’t know the truly relevant docs.
Rocchio 1971 Algorithm (SMART)

- Used in practice:
  \[ \tilde{q}_m = \alpha\tilde{q}_0 + \beta \frac{1}{|D_r|} \sum_{\tilde{d}_j \in D_r} \tilde{d}_j - \gamma \frac{1}{|D_{nr}|} \sum_{\tilde{d}_j \in D_{nr}} \tilde{d}_j \]

- \( D_r \) = set of known relevant doc vectors
- \( D_{nr} \) = set of known irrelevant doc vectors
  - These are different from \( C_r \) and \( C_{nr} \)!
- \( q_m \) = modified query vector; \( q_0 \) = original query vector; \( \alpha, \beta, \gamma \): weights (hand-chosen or set empirically)

- New query moves toward relevant documents and away from irrelevant documents.
Relevance feedback on initial query

- Initial query
- Revised query

x known non-relevant documents
o known relevant documents
Subtleties to note

- Tradeoff $\alpha$ vs. $\beta$ and $\gamma$: If we have a lot of judged documents, we want a higher $\beta$ and $\gamma$
- Some weights in query vector can go negative:
  - Negative term weights are ignored (set to 0)
- **Positive** feedback is more valuable than negative feedback (so, set $\gamma < \beta$; e.g. $\gamma = 0.25$, $\beta = 0.75$) - many systems only allow positive feedback ($\gamma=0$)
- Relevance feedback can improve recall and precision
- Relevance feedback is most useful for increasing recall in situations where recall is important – why?
  - Users can be expected to review results and to take time to iterate – when recall is important.
Relevance Feedback: Assumptions

- 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.
Relevance Feedback: Problems

- 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
- Users are often reluctant to provide explicit feedback
- It’s often harder to understand why a particular document was retrieved after applying relevance feedback
- Information needs may change during the interaction (so what?).
Evaluation of relevance feedback strategies

- Use $q_0$ and compute precision and recall graph
- Use $q_m$ and compute precision recall graph
  - 1) Assess on all documents in the collection
    - Spectacular improvements, but ... it’s cheating!
    - Known relevant documents ranked higher
    - Must evaluate with respect to documents not seen by user
  - 2) Use documents in residual collection (all docs minus those assessed relevant)
    - Measures usually then lower than for original query
    - But a more realistic evaluation
    - Relative performance can be validly compared
- Empirically, one round of relevance feedback is often very useful - two rounds is sometimes marginally useful.
Evaluation of relevance feedback

- Second method – assess only the docs not rated by the user in the first round
  - Could make relevance feedback look worse than it really is
  - Can still assess relative performance of algorithms

- Most satisfactory – use two collections each with their own relevance assessments (i.e., split randomly the collection in two parts)
  - $q_0$ and user feedback from first collection
  - $q_m$ run on second collection and measured.
Evaluation: Caveat

- True evaluation of usefulness must compare to other methods taking the same amount of time – or using similar user effort
- Alternative to relevance feedback: user revises and resubmits query
  - See next topic: query expansion
- 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 – it may be in some "situations".

23
Managing implicit feedback in information search

What can I derive from the fact that a user clicked on the 2nd link?
Learning to rank

- If all the users click on the third link then Yahoo should provide a different ranking for that query.
Adapting the Search Engine

- SE could be adapted to a specific **user**: using just his own clicks (*personalization*)

- SE could be adapted to a community: e.g., the students attending this course

- SE could be adapted to a specific documents collection
Search Engines Bias Users I

- Percentage of queries where a user **viewed** the search result presented at a particular rank (measured with eye tracking).
Search Engines Bias Users II

- **Blue** is the normal rank.
- **Red** is obtained by swapping the top two results.

Percentage of queries where a user clicked the result presented at a given rank.
Query Expansion

- In relevance feedback, users give additional input (relevant/non-relevant) on documents, which is used to reweight terms in the query for documents.

- In **query expansion**, users give additional input (good/bad search term) on **words or phrases**.
  - Generally it is simpler than relevance feedback.
Query assist

Would you expect such a feature to increase the query volume at a search engine?
How do we augment the user query?

- Manual thesaurus
  - E.g. MedLine: **physician**, syn: *doc, doctor, MD, medico*
  - Can be related queries rather than just synonyms

- **Global Analysis:** static; based on all documents in collection
  - Automatically derived thesaurus
    - co-occurrence statistics
  - Refinements based on query log mining
    - Common on the web

- **Local Analysis:** dynamic
  - Analysis of documents in **result set**
Example of manual thesaurus
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 \( \rightarrow \) feline cat
- May weight added terms less than original query terms
- **Generally increases recall**
- Widely used in many science/engineering fields
- May significantly decrease precision, particularly with ambiguous terms
  - “interest rate” \( \rightarrow \) “interest rate benefit evaluate”
- There is a high cost of manually producing a thesaurus
  - And for updating it for scientific changes.
WordNet

- **WordNet®** is a large lexical database of English *(there are also other languages)*
- Nouns, verbs, adjectives and adverbs are grouped into sets of cognitive synonyms - **Synsets** - each expressing a distinct concept
- Synsets are interlinked by means of conceptual-semantic and lexical relations.

<table>
<thead>
<tr>
<th>Relation</th>
<th>Definition</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hyponym</td>
<td>From concepts to superordinates</td>
<td>water$^1 \rightarrow$ liquid</td>
</tr>
<tr>
<td>Hyponym</td>
<td>From concepts to subtypes</td>
<td>water$^1 \rightarrow$ seawater</td>
</tr>
<tr>
<td>Has-Part</td>
<td>From groups to their members</td>
<td>water$^1 \rightarrow$ oxygen</td>
</tr>
<tr>
<td>Part-of</td>
<td>From members to their groups</td>
<td>water$^1 \rightarrow$ ice</td>
</tr>
<tr>
<td>Antonym</td>
<td>Opposites</td>
<td>leader $\rightarrow$ follower</td>
</tr>
</tbody>
</table>
Automatic Thesaurus Generation

- Attempt to generate a thesaurus automatically by analyzing the collection of documents
- Fundamental notion: **similarity between two words** (can we use Jaccard on word bigrams or Levenshtein? )
- Definition 1: Two words are similar if they often co-occur
- 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 based is more robust, grammatical relations are more accurate.
Co-occurrence Thesaurus

- Simplest way to compute a thesaurus is to observe the term-term similarities in $C = AA^T$:
  - $A$ is term-document matrix
  - Alternatively $A = [w_{i,j}]_{M \times N} = (\text{row normalized})$ weight for $(t_i,d_j)$

For each $t_i$, pick terms $t_j$ with high $C_{ij}$ values

What does $C = AA^T$ contain if $A$ is the term-doc incidence (0/1) matrix?
## Automatic Thesaurus Generation Example

<table>
<thead>
<tr>
<th>word</th>
<th>ten nearest neighbors</th>
</tr>
</thead>
<tbody>
<tr>
<td>absolutely</td>
<td>absurd whatsoever totally exactly nothing</td>
</tr>
<tr>
<td>bottomed</td>
<td>dip copper drops topped slide trimmed slight</td>
</tr>
<tr>
<td>captivating</td>
<td>shimmer stunningly superbly plucky witty</td>
</tr>
<tr>
<td>doghouse</td>
<td>dog porch crawling beside downstairs gazed</td>
</tr>
<tr>
<td>Makeup</td>
<td>repellent lotion glossy sunscreen Skin gel</td>
</tr>
<tr>
<td>mediating</td>
<td>reconciliation negotiate cease conciliation</td>
</tr>
<tr>
<td>keeping</td>
<td>hoping bring wiping could some would other</td>
</tr>
<tr>
<td>lithographs</td>
<td>drawings Picasso Dali sculptures Gauguin</td>
</tr>
<tr>
<td>pathogens</td>
<td>toxins bacteria organisms bacterial parasite</td>
</tr>
<tr>
<td>senses</td>
<td>grasp psyche truly clumsy naive innate aw...</td>
</tr>
</tbody>
</table>
Automatic Thesaurus Generation

Discussion

- Quality of associations is usually a problem
- Term ambiguity may introduce irrelevant statistically correlated terms:
  - “Apple computer” → “Apple red fruit computer” (synsets are not distinguished)
- Problems:
  - False positives: Words deemed similar that are not
  - False negatives: Words deemed dissimilar that are similar
- Since terms are highly correlated anyway, expansion may not retrieve many additional documents.
Query assist

- Generally done by query log mining
- Recommend frequent recent queries that contain partial string typed by user
- A ranking problem! View each prior query as a doc – Rank-order those matching partial string ...
Resources

IIR Ch 9