Query Refinement and Search
Result Presentation
(Short) Queries & Information Needs

• A query can be a poor representation of the information need
  – Short queries are often used in search engines due to the limitation of search engines – 2-3 keywords per query on average
  – Would you ask “tropical fish” or “fish” to a person if you want to ask what types of tropical fish were easy to care of?

• A user is expected to think of a few keywords that are likely to be associated with the information need – a huge burden on users
Query-based Stemming

• If the words in documents are stemmed during indexing, the words in queries must also be stemmed
  – “fish village” and “fishing village” are very different! (please try them out)

• Idea: do not stem documents in indexing, instead, expand the query using appropriate word variants
Stem Classes

• Every stemming algorithm (implicitly) generates stem classes
  – A stem class is a group of words that will be transformed into the same stem

• However, stem classes may have errors
  – “Police” and “policy” are semantically different
  – “Banked” is seldom used in finance, but “banking” does

A stem classes by Porter Stemmer
/polic polical polically policely police policeable policed
- policement policer policers polices policial
- policically policier policiers policies policing
- policization policize policly policy policying policys
Word Co-occurrence Analysis

- Assumption: word variants that could substitute for each other should co-occur often in documents

- Method
  - For all pairs of words in a stem class, count how often they co-occur in text windows of $W$ words, where $W$ is typically in the range 50-100
  - Compute a co-occurrence or association metric for each pair, such as Dice’s coefficient
  - Construct a graph where the vertices represent words and the edges are between words whose co-occurrence metric is above a threshold
  - Each connected component of the graph is a new stem class
Term Association Measures

- Dice’s coefficient \( \frac{2n_{ab}}{n_a + n_b} \approx \frac{n_{ab}}{n_a + n_b} \)
  - \( \approx \) means rank-equivalent

- Mutual information \( \log \frac{P(a,b)}{P(a)P(b)} = \log N \frac{n_{ab}}{n_a n_b} \approx \frac{n_{ab}}{n_a n_b} \)
  - It tends to favor infrequent terms
  - Example: \( n_a = n_b = 10, n_{ab} = 5 \), the measure is \( 5 \times 10^{-2} \)
  - \( n_a = n_b = 1000, n_{ab} = 500 \), the measure is \( 5 \times 10^{-4} \)
  - Expected mutual information measure which favors frequent terms

\[
P(a,b) \log \frac{P(a,b)}{P(a)P(b)} = \frac{n_{ab}}{N} \log N \frac{n_{ab}}{n_a n_b} \approx n_{ab} \log N \frac{n_{ab}}{n_a n_b}
\]
Pearson’s Chi-Squared Measure

- Compare the number of co-occurrences with the expected number of co-occurrences if the two words were independent
  - May favor infrequent words

\[
\frac{(n_{ab} - N \frac{n_a n_b}{N N})^2}{N \frac{n_a n_b}{N N}} \approx \frac{(n_{ab} - \frac{n_a n_b}{N})^2}{n_a n_b}
\]

<table>
<thead>
<tr>
<th>Measure</th>
<th>Formula</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mutual information</td>
<td>(\frac{n_{ab}}{n_a n_b})</td>
</tr>
<tr>
<td>(MIM)</td>
<td></td>
</tr>
<tr>
<td>Expected Mutual Information</td>
<td>(n_{ab} \cdot \log(N \cdot \frac{n_{ab}}{n_a n_b}))</td>
</tr>
<tr>
<td>(EMIM)</td>
<td></td>
</tr>
<tr>
<td>Chi-square</td>
<td>(\frac{(n_{ab} - \frac{1}{N} n_a n_b)^2}{n_a n_b})</td>
</tr>
<tr>
<td>((\chi^2))</td>
<td></td>
</tr>
<tr>
<td>Dice’s coefficient</td>
<td>(\frac{n_{ab}}{n_a + n_b})</td>
</tr>
<tr>
<td>(Dice)</td>
<td></td>
</tr>
</tbody>
</table>
Applications

- Word co-occurrence analysis can be applied on query log as well
- Stemming using word co-occurrence analysis does not need a dictionary
  - Learning a dictionary from the document collection or query log
Noisy Channel Model

• Intuition: a person chooses a word $w$ based on a probability distribution $P(w)$, but the noisy channel (presumably her/his brain) causes the person to write the word $e$ instead with probability $P(e|w)$
  – $P(w)$ – the language model, the frequency of occurrence of a word in text
  – $P(e|w)$ – error model, the frequency of different types of spelling errors
  – Collect models from document collection

• Usage: suggesting corrections according to $P(w|e)$
  – $P(w|e) = P(e|w) \times P(w) / P(e)$
Run-on Errors and Context

- Run-on errors: query word boundaries are skipped or mistyped
  - Example: “mainsourcebank” should be “main source bank”
- Mixture of the probability that the word occurs in text and the probability $P(w|w')$ that it occurs following the previous word $w'$
  - The language model probability of $w$ is $\lambda P(w) + (1 - \lambda) P(w|w')$
  - Example: $P(\text{tank}|\text{fish}) >> P(\text{tank}|\text{fresh}) \Rightarrow$ correct “fesh tank” to “fish tank” instead of “fresh tank”
An Iterative Spell Checking Process

• Method
  – Tokenize the query
  – For each token, a set of alternative words and pairs of words are found from both the query log and a trusted dictionary using an edit distance modified by weighting certain types of errors
  – Use the noisy channel model to select the best correction
  – Repeat the process of looking for alternatives and finding the best correction until no better correction is found

• Example: “miniture golfcurses” → “miniature golfcourses” → “miniature golf courses”
Query Expansion

- Using a thesaurus to find synonyms and related words that can be used to expand an initial query
- Example: the NIH Medical Subject Headings thesaurus

<table>
<thead>
<tr>
<th>MeSH Heading</th>
<th>Neck Pain</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tree Number</td>
<td>C10.597.617.576</td>
</tr>
<tr>
<td>Tree Number</td>
<td>C23.888.592.612.553</td>
</tr>
<tr>
<td>Tree Number</td>
<td>C23.888.646.501</td>
</tr>
<tr>
<td>Entry Term</td>
<td>Cervical Pain</td>
</tr>
<tr>
<td>Entry Term</td>
<td>Neckache</td>
</tr>
<tr>
<td>Entry Term</td>
<td>Anterior Cervical Pain</td>
</tr>
<tr>
<td>Entry Term</td>
<td>Anterior Neck Pain</td>
</tr>
<tr>
<td>Entry Term</td>
<td>Cervicalgia</td>
</tr>
<tr>
<td>Entry Term</td>
<td>Cervicodynia</td>
</tr>
<tr>
<td>Entry Term</td>
<td>Neck Ache</td>
</tr>
<tr>
<td>Entry Term</td>
<td>Posterior Cervical Pain</td>
</tr>
<tr>
<td>Entry Term</td>
<td>Posterior Neck Pain</td>
</tr>
</tbody>
</table>
Relevance Feedback

- Asking the user to give feedback on the relevance of documents in an initial set of results
  - The user issues a (short, simple) query
  - The system returns an initial set of retrieval results
  - The user marks some returned documents as relevant or irrelevant
  - The system computes a better representation of the information need based on the user feedback
  - The system displays a revised set of retrieval results
- Relevance feedback can be iterative
A Naïve Algorithm

• Goal: find a query vector \( \mathbf{q} \) which maximizes the similarity between \( \mathbf{q} \) and the relevant documents and minimizes the similarity between \( \mathbf{q} \) and the irrelevant documents.

• Let \( C_r \) and \( C_{ir} \) be the sets of relevant documents and irrelevant documents, respectively.

\[
\mathbf{q}_{opt} = \arg \max_{\mathbf{q}} \left\{ \frac{1}{|C_r|} \sum_{d \in C_r} \text{sim}(\mathbf{q}, d) - \frac{1}{|C_{ir}|} \sum_{d \in C_{ir}} \text{sim}(\mathbf{q}, d) \right\}
\]

\[
= \frac{1}{|C_r|} \sum_{d \in C_r} \mathbf{d} - \frac{1}{|C_{ir}|} \sum_{d \in C_{ir}} \mathbf{d}
\]
Example

- The naïve algorithm is not very useful since the full set of relevant documents is not known

[Diagram showing optimal query, non-relevant documents marked with 'X', and relevant documents marked with 'O']
The Ricchio Algorithm

- Iteratively approach the optimal query using partial relevance feedback
  
  $$\tilde{q}_m = \alpha \tilde{q}_0 + \beta \frac{1}{|D_r|} \sum_{d \in D_r} \hat{d} - \gamma \frac{1}{|D_{ir}|} \sum_{d \in D_{ir}} \hat{d}$$

- If we have many samples, use large $\beta$ and $\gamma$
Relevance Feedback – When?

• The initial query cannot be too bad – it should be close to the documents the user wants
  – Cases relevance feedback does not work (well): misspellings, cross-language information retrieval, mismatch of searcher’s vocabulary versus collection vocabulary

• The relevant documents should be similar to each other
  – Cases relevance feedback does not work (well) where there are multiple clusters caused by subsets of documents using different vocabulary, answers are inherently disjunctive, and answers are instances of general concepts

• Users may be reluctant to provide feedback

• Relevance feedback may also lead to long queries which may not be effective and efficient
Indirect Relevance Feedback

• If a user clicks a link in a list of answers, heuristically, the user is likely interested in the document – can be regarded as a positive feedback
  – Assumption: the snippet is informative enough

• Using clickthrough data
Click-Through Bipartite

A query is represented by a feature vector of URLs.

\[ q_i[j] = \begin{cases} \frac{w_{ij}}{\sqrt{\sum_{e_{ik}} w_{ik}^2}} & \text{if } e_{ik} \text{ exists} \\ 0 & \text{otherwise} \end{cases} \]

Distance between two queries.

\[ \text{dist}(q_i, q_j) = \sqrt{\sum_{u_k} (q_i[k] - q_j[k])^2} \]
Pseudo Relevance Feedback

• When user feedback is unavailable, assume that the top-ranked documents are relevant
  – Words that are frequently used in those documents may be used to expand the initial query
• Example: query “topical fish”
Top-10 Results

- The most frequent terms in the snippets of those top-10 documents except for the stop words may be used to expand the query

  - Tropical, fish, aquarium, freshwater, breeding, information, species, tank, Bandman’s, page, hobby, forum

1. **Badmans Tropical Fish**
   A freshwater aquarium page covering all aspects of the tropical fish hobby. ...
   Notes on a few species and a gallery of photos of African cichlids.
2. **Tropical Fish**
   Info on tropical fish and tropical aquariums, large fish species index with ... Here you will find lots of information on Tropical Fish and Aquariums. ...
3. **The Tropical Fish Homepage - Tropical Fish and Aquariums**
   Offers a range of aquarium products, advice on choosing species, feeding, and health care, and a discussion board.
4. **Tropical Fish Forum**
   Home page for Tropical Fish Internet Directory ... stores, forums, clubs, fish facts, tropical fish compatibility and aquarium ...
5. **Breeding tropical fish**
   ... interested in keeping and/or breeding Tropical, Marine, Pond and Coldwater fish. ...
   Breeding Tropical Fish ... breeding tropical, marine, coldwater & pond fish. ...
6. **Fishlore**
   Includes tropical freshwater aquarium how-to guides, FAQs, fish profiles, articles, and forums.
7. **Cathy’s Tropical Fish Keeping**
   Information on setting up and maintaining a successful freshwater aquarium.
8. **Tropical Fish Place**
   Tropical Fish information for your freshwater fish tank ... great amount of information about a great hobby, a freshwater tropical fish tank. ...
Context and Personalization

• When a query is asked by different people, should the answer be different?
  – Personalization: making answers meeting users’ personal interest
  – User profile: using historical data (e.g., the queries the user asked before, the web pages the user visited before)
  – So far, no good methods

• When a query is asked at different time, should the answer be different?
  – Context-aware: determining the information need of a query using the previous queries (in the same session)
  – A very challenging problem
Search Intent and Context

- User query “gladiator”

- If the user asks query “beautiful mind” before “gladiator”
  - She/he is likely to be interested in the film “gladiator”
  - She/he is likely to be searching the films by Russell Crowe
Challenges and Strategies

• Users often formulate different queries for the same search intent
  – Queries “Microsoft Research Asia”, “MSRA”, “MSR Asia”, and “Microsoft Research Beijing” are similar

• Idea: clustering similar queries into concepts, and representing context by a short sequence of concepts

• Challenge: impractical to scan logs online
  – Mine frequent contexts offline
  – Construct a concept sequence suffix tree for fast online lookup
Framework

Offline part: model learning
- Summarizing queries into concepts by clustering click-through bipartite
- Mining frequent patterns from session data and building a concept sequence suffix tree
Local Search

• Using geographic information derived from the query or from the location of the device that the query comes from to modify the ranking of search results
  – Query “hiking” by a user at Vancouver should return the popular hiking trails at greater Vancouver area

• Method
  – Identify the geographic region associated with web pages using location meta data or automatic detection of location in web page content
  – Identify the geographic region associated with the query using query logs
  – Rank web pages using a comparison of the query and document location information in addition to the usual text- and link-based features
Result Snippets

• A ranked list of document summaries (snippets)
• Links to the actual documents or web pages, cached versions of the page
• Advertisements of short descriptions and links
• Query words are highlighted

**Tropical Fish**
One of the U.K.s Leading suppliers of Tropical, Coldwater, Marine Fish and Invertebrates plus... next day fish delivery service ...

[www.tropicalfish.org.uk/tropical_fish.htm](http://www.tropicalfish.org.uk/tropical_fish.htm)  [Cached page](http://www.tropicalfish.org.uk/tropical_fish.htm)
Snippet Generation

- Text summarization: summarize a document such as a news story
- Query-independent / static summaries and query-dependent / dynamic summaries
  - Snippets are query-dependent in search engines
- Luhn’s approach
  - Intuition: a sentence is important if many important words appear in the sentence, i.e., the “density” of important words is high
  - Rank each sentence in a document using a significance factor – computed based on the occurrences of significant words
  - Select the top sentences for the summary
Luhn’s Approach – Details

• Significant words: words of medium frequency in the document – the frequency is between predefined high frequency and low frequency thresholds

\[
TF(w, d) = \begin{cases}
7 - 0.1 \times (25 - s_d) & \text{if } s_d < 25 \\
7 & \text{if } 25 \leq s_d \leq 40 \\
7 + 0.1 \times (25 - s_d) & \text{otherwise}
\end{cases}
\]
s_d is the number of sentences in the document

• Significant words span: a subsequence in a sentence where significant words are not separated by gaps of insignificant words over a certain length (typically set to 4)

• Significance factor: the square of the number of the significant words in the maximal significant word span divided by the number of words in the maximal significant word span
Example

 initially sentence

 identify significant words

 text span bracketed by significant words

 significance factor = $4^2 \div 7 = 2.3$
Other Factors

- Whether the sentence is a heading, the first two lines of the document
- The total number of query terms occurring in the sentence
- The number of unique query terms in the sentence
- The longest contiguous run of query words in the sentence
- Sentences from the metadata, e.g., `<meta name = "description" content = …>`
- Using a weighted combination of features to rank sentences
Some Guidelines

• Whenever possible, all of the query terms should appear in the summary
  – When query terms are present in the title, however, they do not need to be repeated in the snippet

• Highlighting the query terms present in the URL

• Readable prose in snippets such as complete or near-complete sentences instead of lists of keywords and phrases
  – Readability should be included in the ranking of candidate snippets
Advertising and Sponsored Search

- Advertising presented with search results
- An over-simplified model
  - A search engine maintains a database of advertisements – an ad is associated with a short text description and a link to a web page
  - A query is applied on the advertisement database to find the most relevant ads
- The most relevant ads may not be the most profitable one for search engines
  - Different advertisers may want to pay differently for ads
  - Users click ads not only based on relevance to queries
Keyword Bidding

- Advertisers bid for keywords matching a query – an important factor in ad selection
- Ad popularity is aggregated from web log clickthrough data
  - Query-specific popularity is based on queries that occur on a regular basis
  - General popularity is based on all queries, and can be used for long-tail queries (queries of low frequencies)
Keyword Matching on Ads

- Advertisements are often much shorter than web documents.
- The number of ads in the advertisement database is orders of magnitude smaller than the web page collection.
- Vocabulary mismatch problem: it is important that variations of advertisement keywords that occur in queries are matched.
  - “aquarium” and “fish tanks” should be regarded as matching.
  - Expanding both ad documents and queries may be considered.
Advertisement Oriented Suggestions

• About 50% of the queries in a single session are reformulations
  – The user modifies the original query through word replacements, insertions and deletions

• Significant term associations can be used as potential substitutions
  – Given an initial query, a ranked list of query reformulations can be generated that contain matches for advertising keywords
  – Using term association measures to determine significant associations
Pseudo-RF for Advertisements

- Use the ad text or keywords as a query for a web search, expansion words are selected from the highest ranking web pages
  - A pseudo-relevance feedback approach
- The most effective relevance ranking of advertisements
  - Exact matches of the whole query are ranked first
  - Followed by exact matches of the whole query with words replaced by stems
  - Followed by a probabilistic similarity match of the expanded query with the expanded advertisement
Example

Query: “fish tank”

The advertiser bids on the keyword “aquarium”
Contextual Advertising

- Displaying advertisements related to the current page
- Method: extracting keywords from the content of the current page and search for advertisements
  - Using summarization techniques to select the keywords with high significance weight
Clustering Search Results

• A query may have multiple possible information need
  – Java: coffee, an island, a programming language
• Search results can be clustered into subtopics
  – Results in the same subtopic can be scanned quickly for relevance
• Particularly useful for search on mobile device and small-screen device

<table>
<thead>
<tr>
<th>Pictures (38)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Aquarium Fish (28)</td>
</tr>
<tr>
<td>Tropical Fish Aquarium (26)</td>
</tr>
<tr>
<td>Exporter (31)</td>
</tr>
<tr>
<td>Supplies (32)</td>
</tr>
<tr>
<td>Plants, Aquatic (18)</td>
</tr>
<tr>
<td>Fish Tank (15)</td>
</tr>
<tr>
<td>Breeding (16)</td>
</tr>
<tr>
<td>Marine Fish (16)</td>
</tr>
<tr>
<td>Aquaria (9)</td>
</tr>
</tbody>
</table>
Requirements

• Efficiency
  – Clusters should be specific to queries and based on top-ranked documents
  – Using snippets instead of complete documents

• Understandability
  – Each cluster can be summarized using a single word or phrase
  – Monothetic class – each member of a class has the property that defines the class
     • Otherwise, the class is called polythetic
Faceted Classification

- A set of categories organized into a hierarchy together with a set of facets that describe the important properties associated with the category
- Facets are often manually defined
**Example**

<table>
<thead>
<tr>
<th>Category</th>
<th>Subcategories</th>
<th>Discount</th>
<th>Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>Books (7,845)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Home &amp; Garden (2,477)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Apparel (236)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Home Improvement (169)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jewelry &amp; Watches (76)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sports &amp; Outdoors (71)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Office Products (68)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Toys &amp; Games (62)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Everything Else (44)</td>
<td></td>
<td></td>
<td>$0-$24 (1,032)</td>
</tr>
<tr>
<td>Electronics (26)</td>
<td></td>
<td></td>
<td>$25-$49 (394)</td>
</tr>
<tr>
<td>Baby (25)</td>
<td></td>
<td></td>
<td>$50-$99 (797)</td>
</tr>
<tr>
<td></td>
<td>Home &amp; Garden</td>
<td>Up to 25% off (563)</td>
<td>$100-$199 (206)</td>
</tr>
<tr>
<td></td>
<td>Kitchen &amp; Dining</td>
<td>25% - 50% off (472)</td>
<td>$200-$499 (39)</td>
</tr>
<tr>
<td></td>
<td>Furniture &amp; Décor</td>
<td>50% - 70% off (46)</td>
<td>$500-$999 (9)</td>
</tr>
<tr>
<td></td>
<td>Pet Supplies</td>
<td>70% off or more (46)</td>
<td>$1000-$1999 (5)</td>
</tr>
<tr>
<td></td>
<td>Bedding &amp; Bath</td>
<td></td>
<td>$5000-$9999 (7)</td>
</tr>
<tr>
<td></td>
<td>Patio &amp; Garden</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Art &amp; Craft Supplies</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Wireless Access</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Home Appliances</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Vacuums, Cleaning &amp; Storage</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Brand</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>&lt;brand names&gt;</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Seller</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>&lt;vendor names&gt;</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Summary

- Query-based stemming
  - Word co-occurrence analysis and term association measures
- Noisy channel model
- A spell checking framework
- Relevance feedback
  - The Ricchio algorithm, indirect relevance feedback, and pseudo relevance feedback
- Context and personalization
  - Local search
- Snippet generation
- Advertising and sponsored search
- Clustering search results
To-Do-List

• Read Chapter 6
• Exercises in Chapter 6
• Pick a commercial Web search engine and describe how you think the query is matched to the advertisements for sponsored search. Use examples as evidence for your ideas. Do the same thing for advertisements shown with Web pages
• Suppose you have a list of place names. Can you come up with an algorithm to detect place names or locations in queries? Use examples to show how your algorithm works. Please also think about some situations where your algorithm does not work