A (very brief) overview of Information Retrieval

NLP ML Web
Fall 2013
Andrew Rosenberg
TA/Grader: David Guy Brizan
• Manning, Raghavan & Schütze
• Information retrieval (IR) is finding material (usually documents) of an unstructured nature (usually text) that satisfies an information need from within large collections (usually stored on computers).
Information Retrieval
Information Retrieval

• Query:
  • String, Image, Humming, Product

• Retrieved Documents:
  • Web pages, People, Hotels, Images, Songs, Products, Movies, Tweets, etc.
Information Retrieval

• Requirements:
  • Access large collections of data quickly
  • Flexible Matching
  • Ranked Retrieval
F.D.A. Ruling Would Sharply Restrict Sale of Trans Fats

FDA to phase out use of artery-clogging trans fats
F.D.A. Ruling Would Sharply Restrict Sale of Trans Fats

FDA to phase out use of artery-clogging trans fats

<table>
<thead>
<tr>
<th>Term</th>
<th>Posting List</th>
</tr>
</thead>
<tbody>
<tr>
<td>F.D.A.</td>
<td>1:1, ...</td>
</tr>
<tr>
<td>Ruling</td>
<td>1:2, ...</td>
</tr>
<tr>
<td>Would</td>
<td>1:3, ...</td>
</tr>
<tr>
<td>sharply</td>
<td>1:4, ...</td>
</tr>
<tr>
<td>restrict</td>
<td>1:5, ...</td>
</tr>
<tr>
<td>sale</td>
<td>1:6, ...</td>
</tr>
<tr>
<td>of</td>
<td>1:7, 2:6, ...</td>
</tr>
<tr>
<td>trans</td>
<td>1:8, 2:8, ...</td>
</tr>
<tr>
<td>fats</td>
<td>1:9, 2:9, ...</td>
</tr>
<tr>
<td>FDA</td>
<td>2:1, ...</td>
</tr>
<tr>
<td>to</td>
<td>2:2, ...</td>
</tr>
<tr>
<td>phase</td>
<td>2:3, ...</td>
</tr>
<tr>
<td>out</td>
<td>2:4, ...</td>
</tr>
<tr>
<td>use</td>
<td>2:5, ...</td>
</tr>
<tr>
<td>artery-clogging</td>
<td>2:7, ...</td>
</tr>
</tbody>
</table>
Snippet Generation

- Search for “trans”
- Results:
Snippet Generation

• Search for “trans fats”

• Results:

• F.D.A. Ruling Would Sharply Restrict Sale of Trans Fats - NYTimes ...
  New York Times 14 mins ago - The Food and Drug Administration on Thursday proposed measures that would all but eliminate artificial trans fats, the artery clogging...

• FDA to phase out use of artery-clogging trans fats in effort to prevent heart attacks, deaths
  Washington Post - 25 minutes ago
  WASHINGTON — Heart-clogging trans fats have been slowly disappearing from grocery aisles and restaurant menus in the last decade. Now ...
Inverted Indexing

F.D.A. Ruling Would Sharply Restrict Sale of Trans Fats

FDA to phase out use of artery-clogging trans fats

<table>
<thead>
<tr>
<th>Term</th>
<th>Posting List</th>
</tr>
</thead>
<tbody>
<tr>
<td>F.D.A.</td>
<td>1, ...</td>
</tr>
<tr>
<td>Ruling</td>
<td>1, ...</td>
</tr>
<tr>
<td>Would</td>
<td>1, ...</td>
</tr>
<tr>
<td>sharply</td>
<td>1, ...</td>
</tr>
<tr>
<td>restrict</td>
<td>1, ...</td>
</tr>
<tr>
<td>sale</td>
<td>1, ...</td>
</tr>
<tr>
<td>of</td>
<td>1,2, ...</td>
</tr>
<tr>
<td>trans</td>
<td>1,2, ...</td>
</tr>
<tr>
<td>fats</td>
<td>1,2, ...</td>
</tr>
<tr>
<td>FDA</td>
<td>2, ...</td>
</tr>
<tr>
<td>to</td>
<td>2, ...</td>
</tr>
<tr>
<td>phase</td>
<td>2, ...</td>
</tr>
<tr>
<td>out</td>
<td>2, ...</td>
</tr>
<tr>
<td>use</td>
<td>2, ...</td>
</tr>
<tr>
<td>artery-clogging</td>
<td>2, ...</td>
</tr>
</tbody>
</table>

“of”, “to”, “the”
Remove stop words.

Should you care about capitalization?

Tokenization
FDA vs. F.D.A.
artery-clogging vs artery and clogging
Normalization Strategies

- Case Normalization
  - Make every word lowercase
  - Make every sentence initial word lowercase
- Acronym normalization
  - Remove all word internal periods
- Tokenization
  - How should “don’t” and “artery-clogging” be tokenized? What about “O’Neill” and “AR-15”?  
- Lemmatization / Stemming
  - Convert “talked” to “talk”, “colors” to “color”
  - Porter Stemmer, StanfordNLP, OpenNLP, NLTK
Lemmatization / Stemming

- Available tools:
  - OpenNLP, Stanford NLP tools, NLTK
  - Porter stemmer example rules

<table>
<thead>
<tr>
<th>Rules</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSES -&gt; SS</td>
<td>CARESSES -&gt; CARESS</td>
</tr>
<tr>
<td>IES -&gt; I</td>
<td>PONIES -&gt; PONI</td>
</tr>
<tr>
<td>SS -&gt; SS</td>
<td>CARESS -&gt; CARESS</td>
</tr>
<tr>
<td>S -&gt;</td>
<td>CATS -&gt; CAT</td>
</tr>
</tbody>
</table>
Spelling Correction

• Common misspellings
  • britney spears
    • britain spears, britney’s spears, brandy spears, prittany spears
Spelling Correction

- Approaches to spelling correction
  - Isolated Term, Context Based
- Isolated Term:
  - Minimum Edit Distance
  - N-gram overlap
- Context based
  - Compare to common queries, and common corrected queries
  - Use a language model to find the most likely alternate query
    - Flights form Heathrow
    - Flights from Heathrow
    - Flights forum Heathrow
Minimum Edit Distance

• How “far apart” is britney spears from brittany speers?

• Count the minimum number of insert, delete, and substitute operations required to convert one to the other.

  • brittany speers -> britany speers  
    (Delete t)
  
  • britany speers -> britny speers  
    (Delete a)
  
  • britny speers -> britney speers  
    (Insert e)
  
  • britney speers -> britney spears  
    (Substitute a for e)
Minimum Edit Distance

• Dynamic Programming Solution.
• Solve three subproblems.
  • MED(s[:, -1], t) (delete) cost: 1
  • MED(s, t[:, -1]) (insert) cost: 1
  • MED(s[:, -1], t[:, -1]) (substitute) cost: 2
• Find minimum of these costs.
• MED(INTENTION, EXECUTION)
Example MED calculation

<table>
<thead>
<tr>
<th>#</th>
<th>E</th>
<th>X</th>
<th>E</th>
<th>C</th>
<th>U</th>
<th>T</th>
<th>I</th>
<th>O</th>
<th>N</th>
</tr>
</thead>
<tbody>
<tr>
<td>#</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
</tr>
<tr>
<td>I</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>6</td>
<td>7</td>
</tr>
<tr>
<td>N</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>7</td>
<td>8</td>
</tr>
<tr>
<td>T</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>7</td>
<td>8</td>
<td>9</td>
</tr>
<tr>
<td>E</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
</tr>
<tr>
<td>N</td>
<td>5</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>11</td>
</tr>
<tr>
<td>T</td>
<td>6</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>8</td>
<td>9</td>
<td>10</td>
</tr>
<tr>
<td>I</td>
<td>7</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>9</td>
<td>8</td>
<td>9</td>
</tr>
<tr>
<td>O</td>
<td>8</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>11</td>
<td>9</td>
<td>8</td>
<td>9</td>
</tr>
<tr>
<td>N</td>
<td>9</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>11</td>
<td>12</td>
<td>11</td>
<td>10</td>
<td>9</td>
</tr>
</tbody>
</table>
Aside: Other MED uses

- Best Alignment of Sequences
- DNA sequences
- Protein sequences
- Phonetic sequences
- Acoustic sequences
- Translation alignment
• Another spelling correction approach.

• Misspellings have many letters in common. Identify all words with high n-gram overlap with the query.

• Uses an n-gram index

• Post processing to break ties:
  - bord -> {border, boardroom}
Soundex

- Phonetic mapping for spelling correction
- Particularly useful for names
- One common Soundex algorithm

- Keep the first letter
  - A, E, I, O, U, H, W, Y -> 0
  - B, F, P, V -> 1
  - D, T -> 3

- L -> 4
- M, N -> 5
- R -> 6
- Remove repeated digits.
- Remove 0s
- Trim to 4 characters.
Soundex

- Keep the first letter
  - A, E, I, O, U, H, W, Y -> 0
  - B, F, P, V -> 1
  - D, T -> 3
- L -> 4
- M, N -> 5
- R -> 6
- Remove repeated digits.
- Remove 0s
- Trim to 4 characters.

- BRITTANY -> B635
- BRITNEY -> B635
- BRITNY -> B635
Soundex

- Minimum Edit Distance on Phonetic Labels
- Use Letter-to-Sound (Grapheme-to-phoneme) mapping
- “Andrew” -> AE N D R UW
- “Untrue” -> AH N T R UW
- Optional phonetic feature based substitution cost
- E.g. Replacing stops (T, D, K, P, B, G) with each other is less expensive
Vector Space Scoring

• String match between query and document.

• Modify the query to find more relevant documents.
  • Stemming, Spelling correction, etc.

• Measure Correspondence between the query and document.
  • Compare vector space representations of both
• Use your favorite document/term representation.
• Compare distance between $TV_{query}$ and $TV_{document}$
Cosine Similarity

- Compare only the direction of the two term-vectors.
- Since queries are shorter than documents, they will appear quite different.

$$sim(d_1, d_2) = \frac{\vec{V}(d_1) \cdot \vec{V}(d_2)}{||\vec{V}(d_1)|| ||\vec{V}(d_2)||}$$

Or use a length-normalized term vector function
Relevance

- Relevance is not a feasible substitute for indexing (it’s too slow)
- Relevance is calculated over all ‘hits’ returned by the system to present the most relevant results to the user.

Other relevance indicators

- Document Structure
  - title, meta, subject, body, top, bottom?
  - Link structure (PageRank, etc.)
- Personalized Search
- Localized Search
- Time of day
- Temporally-aware search
- Non-redundant with other results.
Relevance

- How do you get Relevance labels from users?

  1. Pay them.
     - Hire annotators. Give them a set of queries and responses. Ask them to label them as relevant or non-relevant.

  2. Ask Volunteers
     - Was this suggestion helpful? Rate your purchase? Did you mean?

  3. Implicit Feedback
     - Monitor User behavior. Convert this to relevance labels.
Relevance

• Binary labels
  • Has anyone marked this document query pair as relevant?
  • Does everyone?

• Continuous labels
  • What is the expected relevance value?
  • average binary relevance
Relevance Feedback

query

- DocumentA (0.29)
- DocumentQ (0.25)
- DocumentD (0.20)
- DocumentB (0.04)
Rocchio (1971) Algorithm

- Method for identifying an “optimal” query representation.
- Feedback provides positive and negative instances for training.

\[ q_{opt} = \arg \max_q \, \text{sim}(q, C_r) - \text{sim}(q, C_{nr}) \]
Rocchio’s Algorithm

- Move the query representation towards relevant documents, and far from irrelevant documents.

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

• The term you’re looking for might be expressed differently.
  • NIH vs. National Institutes of Health
  • Running vs. jogging
  • Abbreviations, Acronyms
  • Synonyms
  • Common misspellings
  • Paraphrases
Query Expansion

- A manually written thesaurus
  - thesaurus.com, merriam webster, Roget’s, Wordnet
- Automatically learned thesaurus
- Query log analysis
Automatic Thesaurus Generation

- Word co-occurrence
  - If two words appear in the same context (document, paragraph, sentence) are likely related.
- Count the top n words.
- Use LSA to identify closely related terms from a term-document matrix.
Latent Semantic Analysis (reminder)

- PCA operates on word covariance matrix
- Consider a Term-document matrix.
- Singular Value Decomposition (very similar to eigenvector decomposition)

\[ X = U \Sigma V^T \]
Query Log Mining

- Identify related words from documents that are relevant to a query.
Query Log Mining

- Learn correlations between query terms and document terms.
- Include top n related document terms into a query.
Machine Learning for Relevance

- How do you calculate relevance?

- Do you actually care about the absolute relevance scores?
  - What does relevance of 0.93 vs. 0.90 vs. 2.0 vs. -6 mean?

- Rather than predicting relevance, IR systems need a correct ranking of returned documents.
Ranking approaches

- **Pointwise**
  - Predict relevance (or ranking) using regression

- **Pairwise**
  - Each pair of training data points are labeled by their relative rank.
    - \( d_i > d_j \) or \( d_i < d_j \)

- **Listwise**
  - Optimize either point- or pairwise objective functions aggregated over a full list.
• F-Measure (Precision and Recall) is calculated over all returned documents.

• Ordering of returned documents don’t matter.

• **Precision@N**

• Specify N — calculates how good the top N documents are.
Mean Reciprocal Rank

- Calculate the mean reciprocal rank of each correct hit.
- Not appropriate effective with multiple correct hits.

\[
\text{MRR} = \frac{1}{|Q|} \sum_{i=1}^{|Q|} \frac{1}{\text{rank}_i}.
\]

<table>
<thead>
<tr>
<th>cat</th>
<th>dog</th>
</tr>
</thead>
<tbody>
<tr>
<td>catalog</td>
<td>puppies</td>
</tr>
<tr>
<td>settlers of</td>
<td>dog the bounty</td>
</tr>
<tr>
<td>catan</td>
<td>hunter</td>
</tr>
<tr>
<td>cats</td>
<td>dogged pursuit</td>
</tr>
<tr>
<td></td>
<td>of the truth</td>
</tr>
<tr>
<td>cathode ray</td>
<td>doggerel</td>
</tr>
<tr>
<td>tube</td>
<td></td>
</tr>
</tbody>
</table>
Average Precision

\[ AveP = \frac{\sum_{k=1}^{n}(P(k) \times rel(k))}{\text{number of relevant documents}} \]

- P(k) is the precision at k
- rel(k) = 1 if the document at rank k is relevant. 0 otherwise.

- Measures the precision and recall tradeoff.
- Mean average precision is calculated over subsequent queries.
Discounted Cumulative Gain

- Cumulative Gain
  - Sum the total relevance of returned documents.
  
  \[ CG = \sum_{i} r_i \]

- Discounted Cumulative Gain
  - Penalize the relevance by rank

  \[ DCG_p = r_1 + \sum_{i=2}^{p} \frac{r_i}{\log_2 i} \]
Normalized Discounted Cumulative Gain

- What’s the maximal value of $DCG_3$ with binary relevance $(0,1)$?

$$1 + \frac{1}{\log_2 2} + \frac{1}{\log_2 3} \approx 2.63$$

- For cross query compatibility, normalize by maximum DCG.

$$nDCG_p = \frac{DCG_p}{\max DCG_p}$$
Pointwise Ranking

• Generate relevance labels
  • Map this to a numeric value

• Use your favorite Regression algorithm
  • Linear/Polynomial Regression
    • Kernelized
  • CART trees
  • etc.
Rank SVM

- Pairwise algorithm

- Goal of the traditional SVM
  - Increase the absolute distance between points and the decision boundary. (margin size)
  - Reorient the decision boundary to increase the margin size

- Goal of the rank SVM
  - Increase the distances between points that have different ranks.
  - Reorient the projection vector to improve ordering
Traditional SVM

- Identify the largest margin decision boundary.
- maximize margin size, minimize loss, subject to N constraints where N is the number of data points
- Solved with quadratic programming solver

\[ \arg\min_{w, \xi, b} \left\{ \frac{1}{2} \|w\|^2 + C \sum_{i=1}^{n} \xi_i \right\} \]

\[ y_i(w \cdot x_i - b) \geq 1 - \xi_i, \quad \xi_i \geq 0 \]
Rank SVM

Optimization Problem 1. (Ranking SVM)

\[
\begin{align*}
\text{minimize:} & \quad V(\vec{w}, \xi) = \frac{1}{2} \vec{w} \cdot \vec{w} + C \sum \xi_{i,j,k} \quad (12) \\
\text{subject to:} & \\
\forall (d_i, d_j) \in r_1^*: & \quad \vec{w} \Phi(q_1, d_i) \geq \vec{w} \Phi(q_1, d_j) + 1 - \xi_{i,j,1} \\
& \quad \cdots \\
\forall (d_i, d_j) \in r_n^*: & \quad \vec{w} \Phi(q_n, d_i) \geq \vec{w} \Phi(q_n, d_j) + 1 - \xi_{i,j,n} \\
\forall i \forall j \forall k: & \quad \xi_{i,j,k} \geq 0 \\
\end{align*}
\]

\[
\begin{align*}
\forall (d_i, d_j) \in r_1^*: & \quad \vec{w} \Phi(q_1, d_i) > \vec{w} \Phi(q_1, d_j) \\
& \quad \cdots \\
\forall (d_i, d_j) \in r_n^*: & \quad \vec{w} \Phi(q_n, d_i) > \vec{w} \Phi(q_n, d_j) \\
\end{align*}
\]

- \( w_1 \) ranking: 1,2,3,4
- \( w_2 \) ranking: 2,3,4,1
- SVM classification on pairwise differences:

\[
\Phi(q_k, d_i) - \Phi(q_k, d_j)
\]
PRank Algorithm

- Listwise ranking
- Data: \((x_1, y_1) \ldots (x_n, y_n)\) where \(y_i\) is an integral rank
  - Larger is better here
- Loss function: “Ranking loss”
  - The number of positions between the true and hypothesized rank.
PRank Algorithm

- Assume a linear classifier with distinct bias terms for each rank

**Initialize:** Set $w^1 = 0$, $b_1^1, \ldots, b_{k-1}^1 = 0$, $b_k^1 = \infty$.

**Loop:** For $t = 1, 2, \ldots, T$

- Get a new rank-value $x^t \in \mathbb{R}^n$.
- Predict $\hat{y}^t = \min_{r \in \{1, \ldots, k\}} \{r : w^t \cdot x^t - b_r^t < 0\}$.
- Get a new label $y^t$.
- If $\hat{y}^t \neq y^t$ update $w^t$ (otherwise set $w^{t+1} = w^t$, $\forall r : b_r^{t+1} = b_r^t$):
  1. For $r = 1, \ldots, k - 1$:
     - If $y^t < r$ Then $y_r^t = -1$
     - Else $y_r^t = 1$.
  2. For $r = 1, \ldots, k - 1$:
     - If $(w^t \cdot x^t - b_r^t)y_r^t \leq 0$ Then $\tau_r^t = y_r^t$
     - Else $\tau_r^t = 0$.
  3. Update $w^{t+1} \leftarrow w^t + (\sum_r \tau_r^t)x^t$.
     For $r = 1, \ldots, k - 1$ update: $b_r^{t+1} \leftarrow b_r^t - \tau_r^t$

**Output:** $H(x) - \min_{r \in \{1, \ldots, k\}} \{r : w^{T+1} \cdot x - b_r^{T+1} < 0\}$.

<table>
<thead>
<tr>
<th>$y^t = 3$</th>
<th>$y_r^t = 4$</th>
<th>$wx-b$</th>
<th>$t_r^t$</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>+1</td>
<td>&lt;0</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>-1</td>
<td>&lt;0</td>
<td>-1</td>
</tr>
<tr>
<td>3</td>
<td>-1</td>
<td>&lt;0</td>
<td>-1</td>
</tr>
<tr>
<td>2</td>
<td>-1</td>
<td>&gt;0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>-1</td>
<td>&gt;0</td>
<td>0</td>
</tr>
</tbody>
</table>
PRank Algorithm

- Assume a linear classifier with distinct bias terms for each rank

\[
y^{t^*} = 3, \quad y^t = 4
\]

<table>
<thead>
<tr>
<th>(y^r)</th>
<th>(y^r_t)</th>
<th>(wx-b)</th>
<th>(t^r)</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>+1</td>
<td>&lt;0</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>-1</td>
<td>&lt;0</td>
<td>-1</td>
</tr>
<tr>
<td>3</td>
<td>-1</td>
<td>&lt;0</td>
<td>-1</td>
</tr>
<tr>
<td>2</td>
<td>-1</td>
<td>&gt;0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>-1</td>
<td>&gt;0</td>
<td>0</td>
</tr>
</tbody>
</table>
Ensembles of Ranking Results

- Rationales
  - Subsequent related searches (refinement) often generate document sets with little overlap.
  - Combination of moderately correlated rankers can improve results
Combination Strategy

- Relevance judgements are not available for all data points from all rankers.

<table>
<thead>
<tr>
<th>Combination</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CombMIN</td>
<td>minimum relevance</td>
</tr>
<tr>
<td>CombMAX</td>
<td>maximum relevance</td>
</tr>
<tr>
<td>CombSUM</td>
<td>sum all relevance scores</td>
</tr>
<tr>
<td>CombANZ</td>
<td>CombSUM / number non-zero “Average Non Zero”</td>
</tr>
<tr>
<td>CombMNZ</td>
<td>CombSUM * number non-zero “Multiply by Non Zero”</td>
</tr>
</tbody>
</table>


Combination Strategy

- Relevance judgements are not available for all data points from all rankers.

<table>
<thead>
<tr>
<th></th>
<th>RankerA</th>
<th>RankerB</th>
<th>RankerC</th>
<th>RankerD</th>
<th>CombMIN</th>
<th>CombMAX</th>
<th>CombSUM</th>
<th>CombANZ</th>
<th>CombMNZ</th>
</tr>
</thead>
<tbody>
<tr>
<td>Doc1</td>
<td>0.8</td>
<td>0.7</td>
<td>0.3</td>
<td>0.7</td>
<td>0.3</td>
<td>0.8</td>
<td>2.5</td>
<td>0.63</td>
<td>10.00</td>
</tr>
<tr>
<td>Doc2</td>
<td>0.7</td>
<td>0.9</td>
<td>0.4</td>
<td>0.4</td>
<td>0.4</td>
<td>0.9</td>
<td>2.4</td>
<td>0.60</td>
<td>9.60</td>
</tr>
<tr>
<td>Doc3</td>
<td>0.5</td>
<td>0.1</td>
<td>0.0</td>
<td>0.2</td>
<td>0.0</td>
<td>0.5</td>
<td>0.8</td>
<td>0.27</td>
<td>2.40</td>
</tr>
<tr>
<td>Doc4</td>
<td>0.4</td>
<td>0.0</td>
<td>0.3</td>
<td>0.0</td>
<td>0.0</td>
<td>0.4</td>
<td>0.7</td>
<td>0.35</td>
<td>1.40</td>
</tr>
<tr>
<td>Doc5</td>
<td>0.0</td>
<td>0.1</td>
<td>0.0</td>
<td>0.8</td>
<td>0.0</td>
<td>0.8</td>
<td>0.9</td>
<td>0.45</td>
<td>1.80</td>
</tr>
<tr>
<td>Doc6</td>
<td>0.0</td>
<td>0.0</td>
<td>0.7</td>
<td>0.0</td>
<td>0.0</td>
<td>0.9</td>
<td>0.7</td>
<td>0.70</td>
<td>0.70</td>
</tr>
</tbody>
</table>
AdaRank

- Application of **Adaptive Boosting** algorithm to Ranking.

- Steps:
  - Identify a Rank loss function
    - AveP, nDCG commonly used
  - Train weak rankers
    - Feature subset or Data subset
  - Learn linear combination weights

\[ f(x) = \sum_{k=1}^{t} \alpha_k h_k(x) \]
AdaRank Algorithm

- For $t = 1..T$:
  - Create a weak ranker based on training data with weights $W$.
  - Set ranker weight based on current aggregate performance.
  - Reweight data points: $(1+x)/(1-x)$.

Mathematical expressions:

$$
\alpha_t = \frac{1}{2} \ln \frac{\sum_i P_t(i)(1 + E(x_i, h_t, y_i))}{\sum_i P_t(i)(1 - E(x_i, h_t, y_i))}
$$

$$
f_t(x) = \sum \alpha_t h_t(x)
$$

$$
P_{t+1}(i) = \frac{\exp\{-E(x_i, f_t, y_i)\}}{\sum_j \exp\{-E(x_j, f_t, y_j)\}}
$$

Graph from google.com
Recommendation Systems

Top Picks for Andrew

Critically-acclaimed Gritty Dramas

Based on your interest in...

Books

New Release
D.O.M.: Rediscovering ... Alex Atala
$49.95 $34.08
Why recommended?

New Release
Jewels by JAR ...
Adrian Sassoon
$49.99 $24.94
Why recommended?

Python for Data Analysis
Wes McKinney
$39.99 $25.96
Why recommended?

The Art of R Programming
Norman S. Matloff
$39.99 $28.35
Why recommended?

Cracking the Coding Interview
Gayle Laakmann McDowell
$39.99 $28.35
Why recommended?

See all recommendations in Books
Collaborative Filtering

• “What really matters is what you like, not what you are like”
  - Nick Hornby, High Fidelity

• Assumption:

  • **Users** are similar based on having similar reactions to **content**

  • To predict how a user will respond to content, look at how **similar** users responded.

• Relevance is subjective.
Collaborative Filtering Algorithm

- “Memory-based”
- $v_i$ - user i’s average vote
- $l_i$ - items voted on by user i

\[
\bar{v}_i = \frac{1}{|l_i|} \sum_{j \in l_i} v_{ij}
\]

- predicted vote of user a on item j

\[
p_{a,j} = \bar{v}_a + \kappa \sum_{i=1}^{n} w(a, i) (v_{i,j} - \bar{v}_i)
\]

- normalizer
- user similarity
- distance from user mean
Calculating User similarity

• Correlation Coefficient

\[ w(a, i) = \frac{\sum_j (v_{a,j} - \overline{v}_a)(v_{i,j} - \overline{v}_i)}{\sqrt{\sum_j (v_{a,j} - \overline{v}_a)^2} \sqrt{\sum_j (v_{i,j} - \overline{v}_i)^2}} \]

• Cosine Similarity

\[ w(a, i) = \sum_j \frac{v_{a,j}}{\sqrt{\sum_{k \in I_a} v_{a,k}^2}} \frac{v_{i,j}}{\sqrt{\sum_{k \in I_i} v_{i,k}^2}} \]