Information Retrieval

Indexing and Representation: The Vector Space Model

- Document represented by a vector of terms
  - Words (or word stems)
  - Phrases (e.g. computer science)
  - Removes words on “stop list”
    - Documents aren’t about “the”
- Often assumed that terms are uncorrelated.
- Correlations between term vectors implies a similarity between documents.

Document Representation What values to use for terms

- Boolean (term present /absent)
- tf (term frequency) - Count of times term occurs in document.
  - The more times a term *t* occurs in document *d* the more likely it is that *t* is relevant to the document.
  - Used alone, favors common words, long documents.
- df (document frequency)
  - The more a term *t* occurs throughout all documents, the more poorly *t* discriminates between documents
- tf-idf (term frequency * inverse document frequency) -
  - High value indicates that the word occurs more often in this document than average.

Vector Representation

- Documents and Queries are represented as vectors.
- Position 1 corresponds to term 1, position 2 to term 2, position t to term t

$$D_i = w_{d_{i1}}, w_{d_{i2}}, \ldots, w_{d_{it}}$$

$$Q = w_{q1}, w_{q2,} \ldots, w_{qt}$$

$$w = 0 \text{ if a term is absent}$$

Document Vectors

Document ids

	nova	galaxy	heat	h’wood	film	role	diet	fur
A	1.0	0.5	0.3					
B	0.5	1.0						
C				1.0	0.8	0.7		
D				0.9	1.0	0.5		
E							1.0	1.0
F							0.9	1.0
G	0.5		0.7			0.9		
H		0.6	1.0	0.3	0.2	0.8		
I				0.7	0.5		0.1	0.3

Assigning Weights

- Want to weight terms highly if they are
  - frequent in relevant documents … BUT
  - infrequent in the collection as a whole

Assigning Weights

- tf x idf measure:
  - term frequency (tf)
  - inverse document frequency (idf)

$$T_k = \text{term } k \text{ in document } D_i$$

$$tf_{ik} = \text{frequency of term } \mathrm{T}_k \text{ in document } D_i$$

$$idf_k = \text{inverse document frequency of term } \mathrm{T}_k \text{ in } C$$

$$N = \text{ total number of documents in the collection } C$$

$$n_k = \text{the number of documents in } C \text{ that contain } \mathrm{T}_k$$

$$idf_k = \log(N / n_k)$$

tf x idf normalization

- Normalize the term weights (so longer documents are not unfairly given more weight)
  - normalize usually means force all values to fall within a certain range, usually between 0 and 1, inclusive.

$$w_{ik} = \frac{tf_{ik} \log(N / n_k)}{\sqrt{\sum_{k=1}^{t} (tf_{ik})^2 [\log(N / n_k)]^2}}$$

$$\text{Now:}$$

$$sim(D_i, D_j) = \sum_{k=1}^{t} w_{ik} * w_{jk}$$

Vector Space Similarity Measure

combine tf x idf into a similarity measure

$$D_i = w_{d_{i1}}, w_{d_{i2}}, \ldots, w_{d_{it}}$$

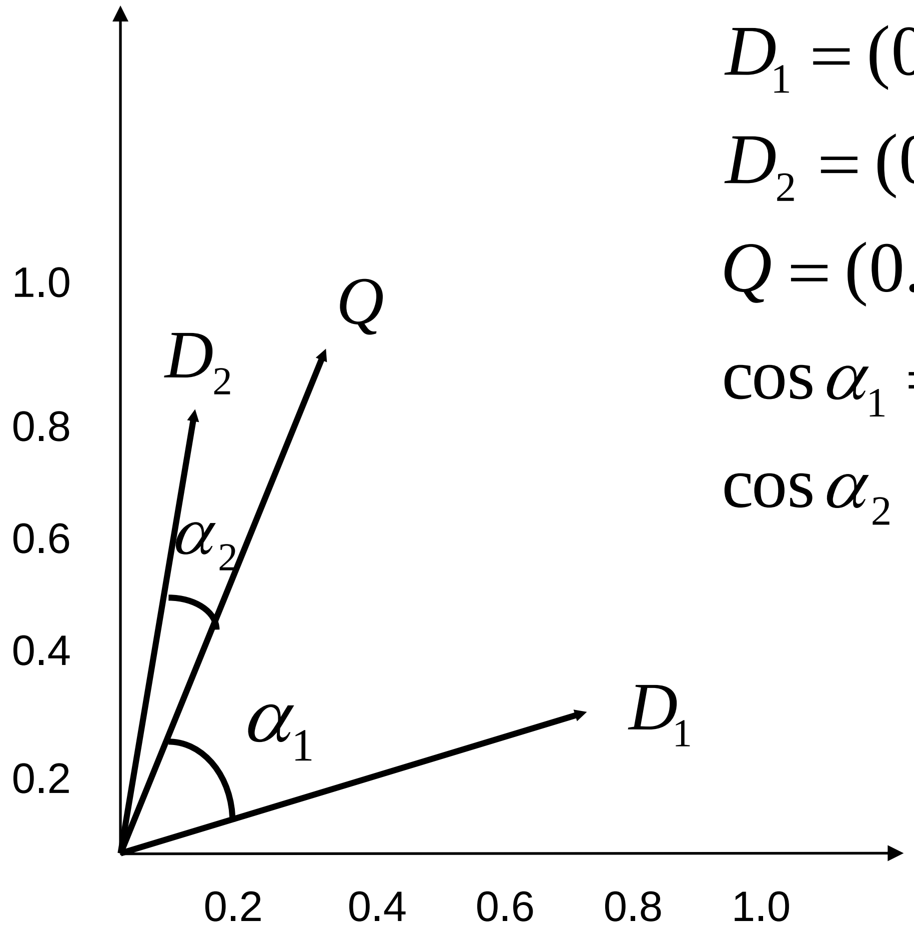
$$Q = w_{q1}, w_{q2}, \ldots, w_{qt} \qquad w = 0 \text{ if a term is absent}$$

$$\text{unnormalized similarity}: \quad sim(Q, D_i) = \sum_{j=1}^{t} w_{qj} * w_{d_{ij}}$$

$$\text{cosine}: \quad sim(Q, D_2) = \frac{\sum_{j=1}^{t} w_{qj} * w_{d_{ij}}}{\sqrt{\sum_{j=1}^{t} (w_{qj})^2 * \sum_{j=1}^{t} (w_{d_{ij}})^2}}$$

(cosine is normalized inner product)

Computing Similarity Scores



$$D_1 = (0.8, 0.3)$$

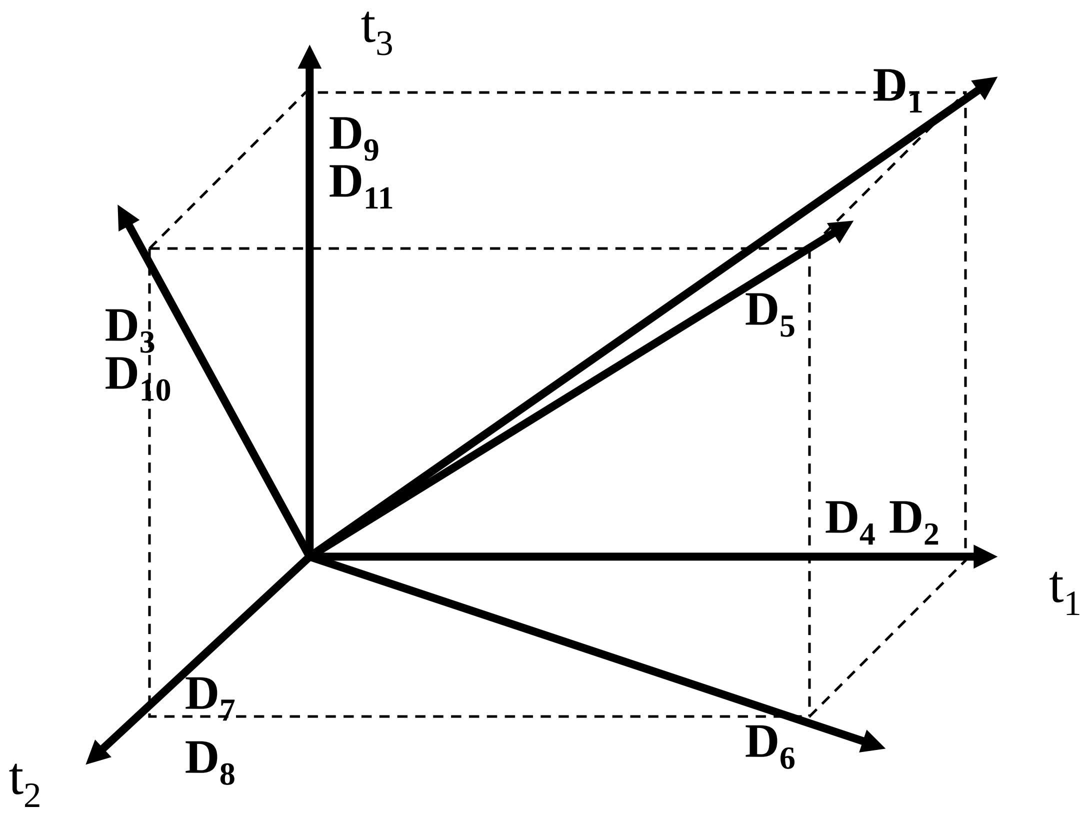
$$D_2 = (0.2, 0.7)$$

$$Q = (0.4, 0.8)$$

$$\cos\alpha_1 = 0.74$$

$$\cos\alpha_2 = 0.98$$

Documents in Vector Space



Computing a similarity score

Say we have query vector $Q = (0.4, 0.8)$

Also, document $D_2 = (0.2, 0.7)$

What does their similarity comparison yield?

$$sim(Q, D_2) = \frac{(0.4 * 0.2) + (0.8 * 0.7)}{\sqrt{[(0.4)^2 + (0.8)^2] * [(0.2)^2 + (0.7)^2]}}$$

$$= \frac{0.64}{\sqrt{0.42}} = 0.98$$

Example

- 假如一篇檔案的總詞語數是100個，而詞語「母牛」出現了3次，那麼「母牛」一詞在該檔案中的詞頻就是3/100=0.03。一個計算檔案頻率（DF）的方法是測定有多少份檔案出現過「母牛」一詞，然後除以檔案集裡包含的檔案總數。所以，如果「母牛」一詞在1,000份檔案出現過，而檔案總數是10,000,000份的話，其逆向檔案頻率就是log（10,000,000 / 1,000）=4。最後的tf-idf的分數為0.03 * 4=0.12。

Similarity Measures

$$|Q \cap D|$$

Simple matching (coordination level match)

$$2\frac{|Q \cap D|}{|Q| + |D|}$$

Dice's Coefficient

$$\frac{|Q \cap D|}{|Q \cup D|}$$

Jaccard's Coefficient

$$\frac{|Q \cap D|}{|Q|^{1/2} \times |D|^{1/2}}$$

Cosine Coefficient

$$\frac{|Q \cap D|}{\min(|Q|, |D|)}$$

Overlap Coefficient

Evaluation

- Relevance
- Evaluation of IR Systems
  - Precision vs. Recall
  - Cutoff Points
  - Test Collections/TREC
  - Blair & Maron Study

What to Evaluate?

- How much learned about the collection?
- How much learned about a topic?
- How much of the information need is satisfied?
- How inviting the system is?

What to Evaluate?

- What can be measured that reflects users' ability to use system? (Cleverdon 66)
  - Coverage of Information
  - Form of Presentation
  - Effort required/Ease of Use
  - Time and Space Efficiency
  - Recall
    - proportion of relevant material actually retrieved
  - Precision
    - proportion of retrieved material actually relevant

effectiveness

Relevance

- In what ways can a document be relevant to a query?
  - Answer precise question precisely.
  - Partially answer question.
  - Suggest a source for more information.
  - Give background information.
  - Remind the user of other knowledge.
  - Others ...

Standard IR Evaluation

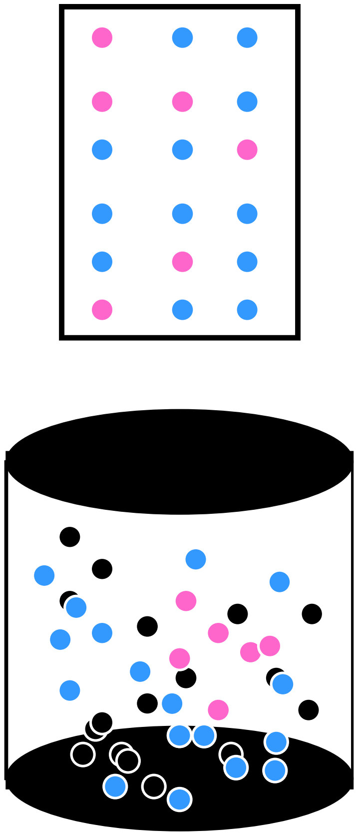
- Precision

$$\frac{\#\ \text{relevant retrieved}}{\#\ \text{retrieved}}$$

- Recall

$$\frac{\#\ \text{relevant retrieved}}{\#\ \text{relevant in collection}}$$

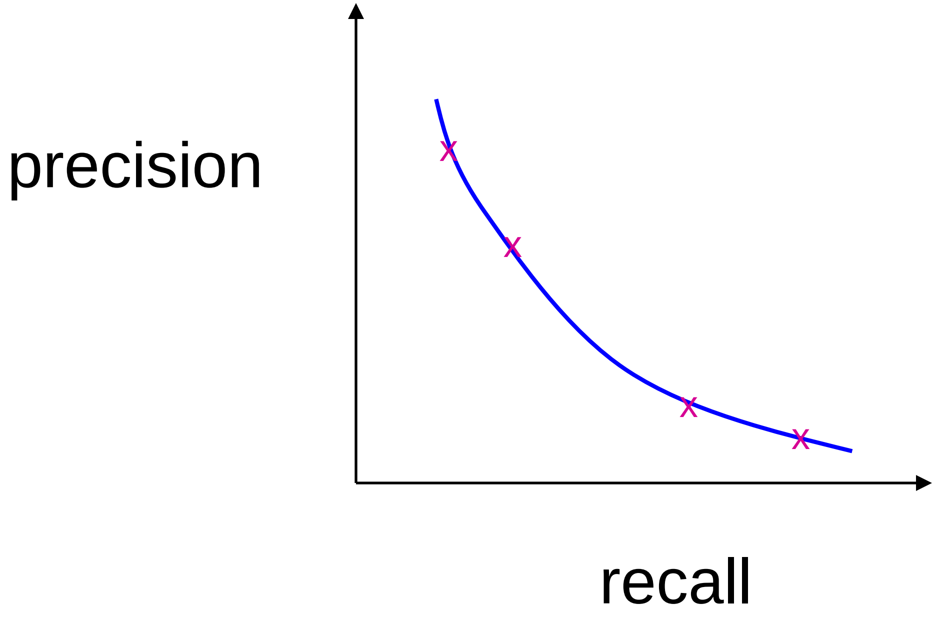
Retrieved Documents



Collection

Precision/Recall Curves

- There is a tradeoff between Precision and Recall
- So measure Precision at different levels of Recall



Precision/Recall Curves

- **Difficult to determine which of these two hypothetical results is better:**