Information Retrieval

INFO 4300 / CS 4300

- Retrieval models
  - Older models
    - » Boolean retrieval
    - » **Vector Space model**
  - Probabilistic Models
    - » **BM25**
    - » Language models
  - Combining evidence
    - » Inference networks
    - » Learning to Rank

Today

Retrieval Models

- Provide a mathematical framework for defining the search process
  - includes explanation of assumptions
  - basis of many ranking algorithms
  - can be implicit
- Progress in retrieval models has corresponded with improvements in effectiveness
- Theories about relevance

Relevance

- Complex concept that has been studied for some time
  - Many factors to consider
  - People often disagree when making relevance judgments
- Retrieval models make various assumptions about relevance to simplify problem
  - e.g., *topical* vs. *user* relevance
  - e.g., *binary* vs. *multi-valued* relevance

Retrieval Model Overview

- Older models
  - Boolean retrieval
  - **Vector Space model**
- Probabilistic Models
  - **BM25**
  - Language models
- Combining evidence
  - Inference networks
  - Learning to Rank

Vector Space Model

- Documents and query represented by a vector of term weights
- Collection represented by a matrix of term weights

$$D_i = (d_{i1}, d_{i2}, \dots, d_{it}) \qquad Q = (q_1, q_2, \dots, q_t)$$

$$\begin{array}{ccccc} & Term_1 & Term_2 & \dots & Term_t \\ Doc_1 & d_{11} & d_{12} & \dots & d_{1t} \\ Doc_2 & d_{21} & d_{22} & \dots & d_{2t} \\ \vdots & \vdots & & & \\ Doc_n & d_{n1} & d_{n2} & \dots & d_{nt} \end{array}$$

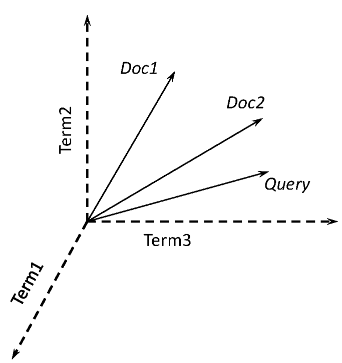
Vector Space Model

D_1 Tropical Freshwater Aquarium Fish.
D_2 Tropical Fish, Aquarium Care, Tank Setup.
D_3 Keeping Tropical Fish and Goldfish in Aquariums, and Fish Bowls.
D_4 The Tropical Tank Homepage - Tropical Fish and Aquariums.

Terms	**Documents**			
	D_1	**D_2**	**D_3**	**D_4**
aquarium	1	1	1	1
bowl	0	0	1	0
care	0	1	0	0
fish	1	1	2	1
freshwater	1	0	0	0
goldfish	0	0	1	0
homepage	0	0	0	1
keep	0	0	1	0
setup	0	1	0	0
tank	0	1	0	1
tropical	1	1	1	2

Vector Space Model

- 3-d pictures useful, but can be misleading for high-dimensional space



Vector Space Model

- Documents ranked by distance between points representing query and documents
  - *Similarity* measure more common than a distance or *dissimilarity* measure
  - e.g. Cosine correlation

$$Cosine(D_i, Q) = \frac{\sum_{j=1}^{t} d_{ij} \cdot q_j}{\sqrt{\sum_{j=1}^{t} d_{ij}{}^2 \cdot \sum_{j=1}^{t} q_j{}^2}}$$

Similarity Calculation

– Consider two documents D_1, D_2 and a query Q

» D_1 = (0.5, 0.8, 0.3), D_2 = (0.9, 0.4, 0.2), Q = (1.5, 1.0,

$$
\begin{aligned}
Cosine(D_1, Q) &= \frac{(0.5 \times 1.5) + (0.8 \times 1.0)}{\sqrt{(0.5^2 + 0.8^2 + 0.3^2)(1.5^2 + 1.0^2)}} \\
&= \frac{1.55}{\sqrt{(0.98 \times 3.25)}} = 0.87
\end{aligned}
$$

$$
\begin{aligned}
Cosine(D_2, Q) &= \frac{(0.9 \times 1.5) + (0.4 \times 1.0)}{\sqrt{(0.9^2 + 0.4^2 + 0.2^2)(1.5^2 + 1.0^2)}} \\
&= \frac{1.75}{\sqrt{(1.01 \times 3.25)}} = 0.97
\end{aligned}
$$

Term Weights

- *tf.idf* weight
  - Term frequency weight measures importance in document: $tf_{ik} = \frac{f_{ik}}{\sum_{j=1}^{t} f_{ij}}$
  - Inverse document frequency measures importance in collection: $idf_k = \log \frac{N}{n_k}$
  - Some heuristic modifications

$$
d_{ik} = \frac{(\log(f_{ik})+1) \cdot \log(N/n_k)}{\sqrt{\sum_{k=1}^{t} [(\log(f_{ik})+1.0) \cdot \log(N/n_k)]^2}}
$$

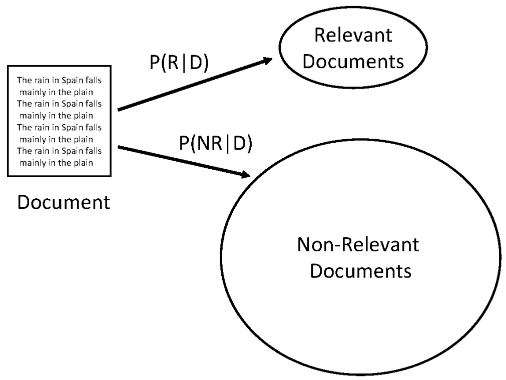
Vector Space Model

- Advantages
  - Simple computational framework for ranking
  - Any similarity measure or term weighting scheme could be used
- Disadvantages
  - Assumption of term independence
  - No *predictions* about techniques for effective ranking

Probability Ranking Principle

- Robertson (1977)
  - "If a reference retrieval system's response to each request is a ranking of the documents in the collection in order of decreasing probability of relevance to the user who submitted the request,
  - where the probabilities are estimated as accurately as possible on the basis of whatever data have been made available to the system for this purpose,
  - the overall effectiveness of the system to its user will be the best that is obtainable on the basis of those data."

IR as Classification



Bayes Classifier

- Bayes Decision Rule
  - A document *D* is relevant if *P*(*R*|*D*) > *P*(*NR*|*D*)
- Estimating probabilities
  - use Bayes Rule

$$P(R|D) = \frac{P(D|R)P(R)}{P(D)}$$

  - classify a document as relevant if

$$\frac{P(D|R)}{P(D|NR)} > \frac{P(NR)}{P(R)}$$

    » lhs is *likelihood ratio*

Estimating P(D|R)

- Assume term independence

$$P(D|R) = \prod_{i=1}^{t} P(d_i|R)$$

- ***Binary independence model***
  - document represented by a vector of *t* binary features indicating term occurrence (or non-occurrence)

BM25

- Popular and effective ranking algorithm based on binary independence model
  - adds document and query term weights

$$\sum_{i \in Q} \log \frac{(r_i+0.5)/(R-r_i+0.5)}{(n_i-r_i+0.5)/(N-n_i-R+r_i+0.5)} \cdot \frac{(k_1+1)f_i}{K+f_i} \cdot \frac{(k_2+1)qf_i}{k_2+qf_i}$$

  - k_1, k_2 and K are parameters whose values are set empirically
  - $K = k_1((1-b) + b \cdot \frac{dl}{avdl})$ *dl* is doc length
  - Typical TREC value for k_1 is 1.2, k_2 varies from 0 to 1000, b = 0.75

- r_i is the # of relevant documents containing term i
- (set to 0 if no relevancy info is known)
- n_i is the # of docs containing term i
- N is the total # of docs in the collection
- R is the number of relevant documents for this query
- (set to 0 if no relevancy info is known)
- f_i is the frequency of term i in the doc under consideration
- qf_i is the frequency of term i in the query
- k_1 determines how the tf component of the term weight changes as f_i increases. (if 0, then tf component is ignored.) Typical value for TREC is 1.2; so f_i is very non-linear (similar to the use of *log f* in term wts of the vector space model) --- after 3 or 4 occurrences of a term, additional occurrences will have little impact.
- k_2 has a similar role for the query term weights. Typical values (see slide) make the equation less sensitive to k_2 than k_1 because query term frequencies are much lower and less variable than doc term frequencies.
- K is more complicated. Its role is basically to normalize the tf component by document length.
- b regulates the impact of length normalization. (0 means none; 1 is full normalization.)

BM25 Example

- Query with two terms, "president lincoln", (*qf = 1)*
- No relevance information (*r and R are* zero)
- *N* = 500,000 documents
- *"president"* occurs in 40,000 documents (n_1 = 40,000)
- *"lincoln"* occurs in 300 documents (n_2 = 300)
- "president" occurs 15 times in doc (f_1 = 15)
- *"lincoln"* occurs 25 times (f_2 = 25)
- document length is 90% of the average length (*dl/avdl* = .9)
- k_1 = 1.2, *b* = 0.75, and k_2 = 100
- *K* = 1.2 · (0.25 + 0.75 · 0.9) = 1.11

BM25 Example

$$
\begin{aligned}
BM25(Q,D) &= \\
&\log \frac{(0+0.5)/(0-0+0.5)}{(40000-0+0.5)/(500000-40000-0+0+0.5)} \\
&\times \frac{(1.2+1)15}{1.11+15} \times \frac{(100+1)1}{100+1} \\
&+ \log \frac{(0+0.5)/(0-0+0.5)}{(300-0+0.5)/(500000-300-0+0+0.5)} \\
&\times \frac{(1.2+1)25}{1.11+25} \times \frac{(100+1)1}{100+1} \\
&= \log 460000.5/40000.5 \cdot 33/16.11 \cdot 101/101 \\
&\quad + \log 499700.5/300.5 \cdot 55/26.11 \cdot 101/101 \\
&= 2.44 \cdot 2.05 \cdot 1 + 7.42 \cdot 2.11 \cdot 1 \\
&= 5.00 + 15.66 = 20.66
\end{aligned}
$$

BM25 Example

- Effect of term frequencies

Frequency of "president"	Frequency of "lincoln"	BM25 score
15	25	20.66
15	1	12.74
15	0	5.00
1	25	18.2
0	25	15.66