![PURDUE UNIVERSITY]

CS54701: Information Retrieval

Retrieval Models

21 January 2016

Prof. Chris Clifton

Indiana Center for Database Systems

16

Retrieval Models: Vector Space Model

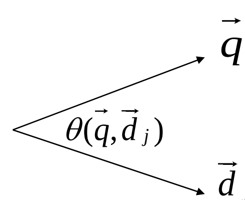
Given two vectors, query and document

10 0 1

- Query: $\vec{q} = (q_1, q_2, \ldots, q_n)$

1 2

- Document: $\vec{d}_j = (d_{j1}, d_{j2}, \ldots, d_{jn})$
- calculate the similarity



Cosine similarity: Angle between vectors

$$sim(\vec{q}, \vec{d}_j) = \cos(\theta(\vec{q}, \vec{d}_j))$$

$$\cos(\theta(\vec{q}, \vec{d}_j)) = \frac{\vec{q} \cdot \vec{d}_j}{\|\vec{q}\| \|\vec{d}\|} = \frac{q_1 d_{j,1} + q_2 d_{j,2} + \ldots + q_j d_{j,n}}{\|\vec{q}\| \|\vec{d}\|} = \frac{q_1 d_{j,1} + q_2 d_{j,2} + \ldots + q_j d_{j,n}}{\sqrt{q_1^2 + \ldots + q_n^2}\sqrt{d_{j1}^2 + \ldots + d_{jn}^2}}$$

Retrieval Models: Vector Space Model

Vector representation

	Java	Sun	Starbucks
D1	1	1	0
D2	1	0	1
D3	1	0	0
Query	1	0.2	1

Similarity Score	D1	D2	D3
Query	0.59	0.99	0.70

Retrieval Models: Vector Space Model

Vector Coefficients

- The coefficients (vector elements) represent term evidence/ term importance
- It is derived from several elements
  - Document term weight: Evidence of the term in the document/query
  - Collection term weight: Importance of term from observation of collection
  - Length normalization: Reduce document length bias
- Naming convention for coefficients:

$$d_{j,k}.q_k = DCL.DCL$$

First triple represents document term; second for query term

Retrieval Models: Vector Space Model

Common vector weight components:

- lnc.ltc: widely used term weight
  - “l”: log(tf+1)
  - “n”: no weight/normalization
  - “t”: log(N/df)
  - “c”: cosine normalization

$$\frac{q_1 d_{j1} + q_2 d_{j2} .. + q_n d_{jn}}{\left\|\vec{q}\right\| \left\|\vec{d}_j\right\|} = \frac{\sum_k \left[\left(\log(tf_q(k)+1)\right)\left(\log(tf_j(k)+1)\right)\log\frac{N}{df(k)} \right]}{\sqrt{\sum_k \left[\left(\log(tf_q(k)+1)\right)^2 \right]} \sqrt{\sum_k \left[\left(\log(tf_j(k)+1)\right)\log\frac{N}{df(k)} \right]^2}}$$

Retrieval Models: Vector Space Model

Common vector weight components:

- dnn.dtb: handle varied document lengths
  - “d”: 1+ln(1+ln(tf))
  - “t”: log((N/df)
  - “b”: 1/(0.8+0.2*docleng/avg_doclen)

Retrieval Models: Vector Space Model

- Standard vector space
  - Represent query/documents in a vector space
  - Each dimension corresponds to a term in the vocabulary
  - Use a combination of components to represent the term evidence in both query and document
  - Use similarity function to estimate the relationship between query/documents (e.g., cosine similarity)

Retrieval Models: Vector Space Model

Advantages:

- Best match method; it does not need a precise query
- Generated ranked lists; easy to explore the results
- Simplicity: easy to implement
- Effectiveness: often works well
- Flexibility: can utilize different types of term weighting methods
- Used in a wide range of IR tasks: retrieval, classification, summarization, content-based filtering…

Retrieval Models: Vector Space Model

Disadvantages:

- Hard to choose the dimension of the vector ("basic concept")
  - Terms may not be the best choice
- Assume independent relationship among terms
- Heuristic for choosing vector operations
  - Choose of term weights
  - Choose of similarity function
- Assume a query and a document can be treated in the same way

Retrieval Models: Vector Space Model

What are good vector representations?

- Orthogonal: the dimensions are linearly independent ("no overlapping")
- No ambiguity (e.g., Java)
- Wide coverage and good granularity
- Good interpretation (e.g., representation of semantic meaning)
- Many possibilities: words, stemmed words, "latent concepts"….

Retrieval Models: Latent Semantic Indexing

Dual space of terms and documents

	C1	C2	C3	C4	B1	B2	B3
information	1	1	0	0	0	0	0
retrieval	1	1	0	0	0	0	0
machine	1	1	1	1	0	0	0
learning	0	1	1	1	0	0	0
system	1	0	1	0	0	0	0
protein	0	0	0	0	0	1	1
gene	0	0	1	0	1	1	0
mutation	0	0	0	0	0	1	1
expression	0	0	0	0	1	0	1

Retrieval Models: Latent Semantic Indexing

Latent Semantic Indexing (LSI): Explore correlation between terms and documents

- Two terms are correlated (may share similar semantic concepts) if they often co-occur
- Two documents are correlated (share similar topics) if they have many common words

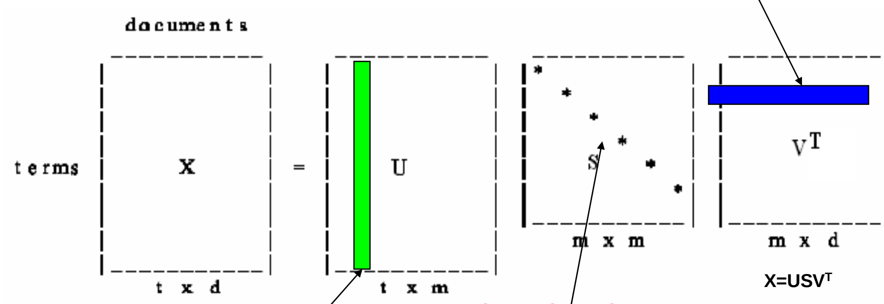
Latent Semantic Indexing (LSI): Associate each term and document with a small number of semantic concepts/topics

Retrieval Models: Latent Semantic Indexing

Using singular value decomposition (SVD) to find a small set of concepts/topics

m: number of concepts/topics

Representation of concept in document space; $V^TV=I_m$



$X=USV^T$

$U^TU=I_m$

$V^TV=I_m$

Diagonal matrix: concept space

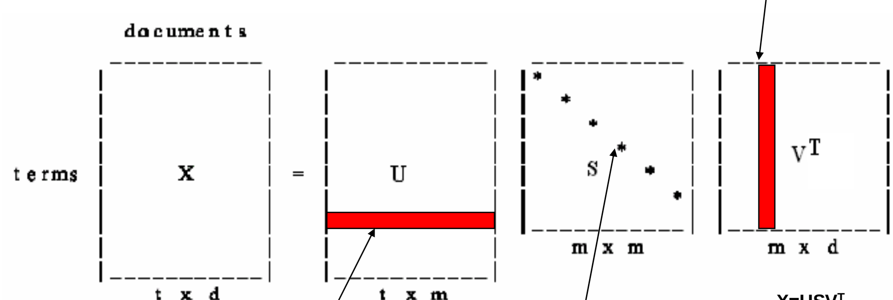
Representation of concept in term space; $U^TU=I_m$

Retrieval Models: Latent Semantic Indexing

Using singular value decomposition (SVD) to find a small set of concepts/topics

m: number of concepts/topics

Representation of document in concept space



$X=USV^T$

$U^TU=I_m$

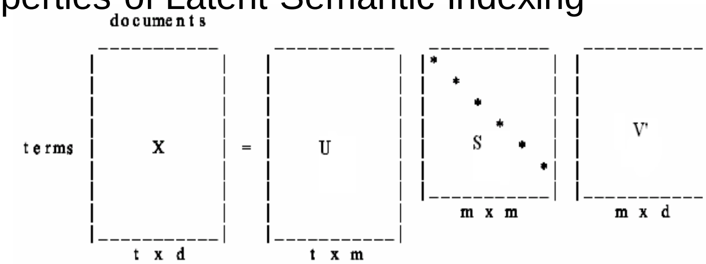
$V^TV=I_m$

Diagonal matrix: concept space

Representation of term in concept space

Retrieval Models: Latent Semantic Indexing

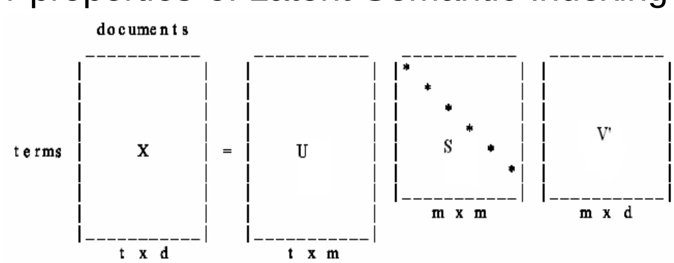
Properties of Latent Semantic Indexing



- Diagonal elements of S as S_k in descending order, the larger the more important
- $\hat{x}_k = \sum_{i \le k} u_k S_k v_k'$ is the rank-k matrix that best approximates X, where u_k and v_k are the column vector of U and V

Retrieval Models: Latent Semantic Indexing

Other properties of Latent Semantic Indexing



- The columns of U are eigenvectors of XX^T
- The columns of V are eigenvectors of X^TX
- The singular values on the diagonal of S, are the positive square roots of the nonzero eigenvalues of both AA^T and A^TA

Retrieval Models: Latent Semantic Indexing

	C1	C2	C3	C4	B1	B2	B3
information	1	1	0	0	0	0	0
retrieval	1	1	0	0	0	0	0
machine	1	1	1	1	0	0	0
learning	0	1	1	1	0	0	0
system	1	0	1	0	0	0	0
protein	0	0	0	0	0	1	1
gene	0	0	1	0	1	1	0
mutation	0	0	0	0	0	1	1
expression	0	0	0	0	1	0	1

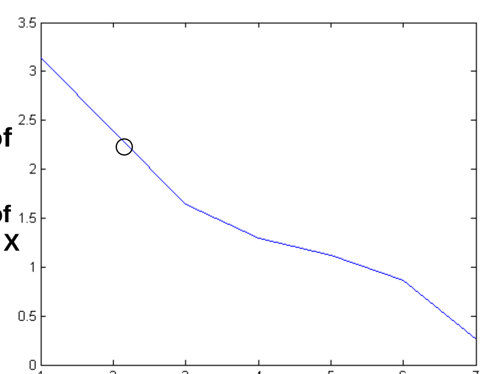
$$\begin{pmatrix} -0.3467 & -0.1369 \\ -0.3467 & -0.1369 \\ -0.6215 & -0.0987 \\ -0.4544 & -0.0327 \\ -0.3329 & -0.0049 \\ -0.0452 & 0.5225 \\ -0.2245 & 0.4859 \\ -0.0452 & 0.5225 \\ -0.0401 & 0.4118 \end{pmatrix} \times \begin{pmatrix} 3.1395 & 0 \\ 0 & 2.3912 \end{pmatrix} \times \begin{pmatrix} -0.5248 & -0.5635 & -0.5202 & -0.3427 & -0.0843 & -0.1003 & -0.0415 \\ -0.1578 & -0.1695 & 0.1462 & -0.0550 & 0.3754 & 0.6402 & 0.6092 \end{pmatrix}$$

Retrieval Models: Latent Semantic Indexing

Importance of concepts

Importance of Concept

Reflect Error of Approximating X with small S



Size of S_k

Retrieval Models: Latent Semantic Indexing

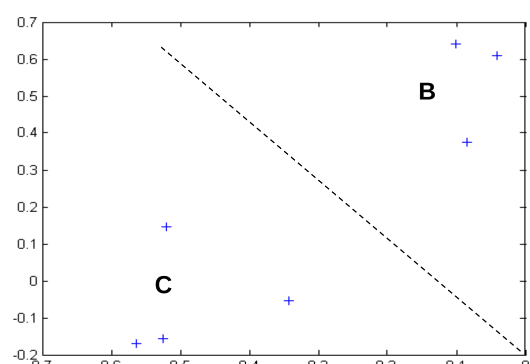
- SVD representation
  - Reduce high dimensional representation of document or query into low dimensional concept space
  - SVD tries to preserve the Euclidean distance of document/term vector

	C1	C2
information	1	1
retrieval	1	1
machine	1	1
learning	0	1
system	1	0
protein	0	0
gene	0	0
mutation	0	0
expression	0	0

	C1	C2
Concept 1	-0.5248	-0.1578
Concept 2	-0.5635	-0.1695

Retrieval Models: Latent Semantic Indexing

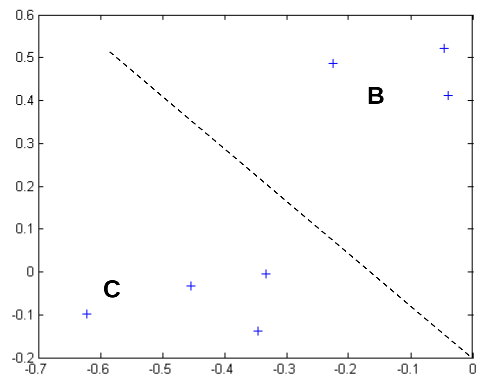
- SVD representation



Representation of the documents in two dimensional concept space

Retrieval Models: Latent Semantic Indexing

- SVD representation



Representation of the terms in two dimensional concept space

Retrieval Models: Latent Semantic Indexing

Retrieval with respect to a query

- Map (fold-in) a query into the representation of the concept space $\vec{q}'^T = \vec{q}^T U_k Inv(S_k)$
- Use the new representation of the query to calculate the similarity between query and all documents
  - Cosine Similarity

Retrieval Models: Latent Semantic Indexing

Query: Machine Learning Protein

	C1	C2	C3	C4	B1	B2	B3
information	1	1	0	0	0	0	0
retrieval	1	1	0	0	0	0	0
machine	1	1	1	1	0	0	0
learning	0	1	1	1	0	0	0
system	1	0	1	0	0	0	0
protein	0	0	0	0	0	1	1
gene	0	0	1	0	1	1	0
mutation	0	0	0	0	0	1	1
expression	0	0	0	0	1	0	1

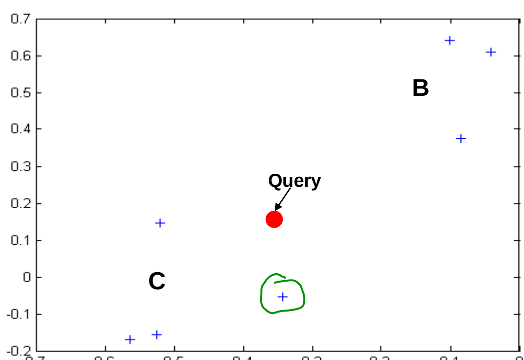
Representation of the query in the term vector space:

$[0\ 0\ 1\ 1\ 0\ 1\ 0\ 0\ 0]^T$

Retrieval Models: Latent Semantic Indexing

Representation of the query in the latent semantic space (2 concepts):

$$\vec{q}'^T = \vec{q}^T U_k Inv(S_k) \quad =[-0.3571\ 0.1635]^T$$



Retrieval Models: Latent Semantic Indexing

Comparison of Retrieval Results in term space and concept space

	C1	C2	C3	C4	B1	B2	B3
information	1	1	0	0	0	0	0
retrieval	1	1	0	0	0	0	0
machine	1	1	1	1	0	0	0
learning	0	1	1	1	0	0	0
system	1	0	1	0	0	0	0
protein	0	0	0	0	0	1	1
gene	0	0	1	0	1	1	0
mutation	0	0	0	0	0	1	1
expression	0	0	0	0	1	0	1
Query Similarity in term space	0.29	0.58	0.58	0.82	0	0.33	0.33
Query Similarity in concept space	0.75	0.75	0.98	0.83	0.61	0.55	0.48

Query: Machine Learning Protein

Retrieval Models: Latent Semantic Indexing

Problems with latent semantic indexing

- Difficult to decide the number of concepts
- There is no probabilistic interpolation for the results
- The complexity of the LSI model obtained from SVD is costly

Retrieval Models: Outline

- Retrieval Models
- Exact-match retrieval method
  - Unranked Boolean retrieval method
  - Ranked Boolean retrieval method
- Best-match retrieval
  - Vector space retrieval method
  - Latent semantic indexing