PURDUE UNIVERSITY | Department of Computer Science

CS47300: Web Information Search and Management

Prof. Chris Clifton

4 September 2020

Material adapted from course created by Dr. Luo Si, now leading Alibaba research group

Indiana Center for Database Systems ™

PURDUE UNIVERSITY Department of Computer Science

Retrieval Models: Vector Space Model

- Any text object can be represented by a term vector
  - Documents, queries, passages, sentences
  - A query can be seen as a short document
- Similarity is determined by distance in the vector space
  - Example: cosine of the angle between two vectors

(Research) Famous Examples

- The SMART system
  - Developed at Cornell University: 1960-1999
  - Still quite popular
- The Lucene system
  - Open source information retrieval library; (Based on Java)
  - Works with Hadoop (Map/Reduce) in large scale app (e.g., Amazon Book)

Retrieval Models: Vector Space Model

Vector space model vs. Boolean model

- Boolean models
  - Query: a Boolean expression that a document must satisfy
  - Retrieval: Deductive inference
- Vector space model
  - Query: viewed as a short document in a vector space
  - Retrieval: Similarity search

Retrieval Models: Vector Space Model

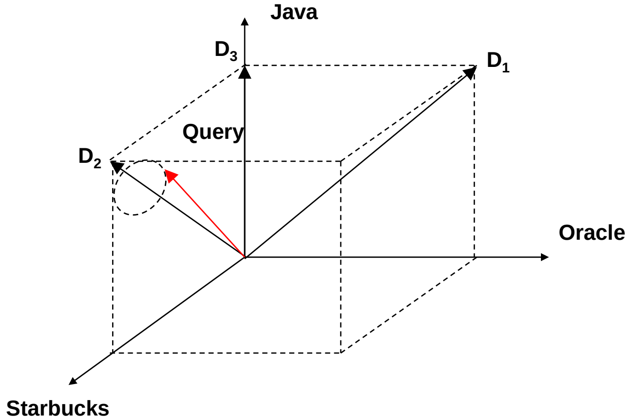
- Vector representation

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

PURDUE UNIVERSITY
Department of Computer Science

Retrieval Models: Vector Space Model

- Vector representation

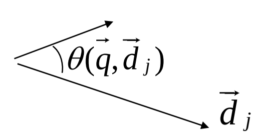


PURDUE UNIVERSITY
Department of Computer Science

Retrieval Models: Vector Space Model

Give two vectors of query and document

- Query $\vec{q} = (q_1, q_2, \dots, q_n)$
- Document $\overrightarrow{d_j} = (d_{j1}, d_{j2}, \dots, d_{jn})$
- calculate the similarity



Cosine similarity: Angle between vectors

$$\text{sim}\left(\vec{q}, \overrightarrow{d_j}\right) = \cos\left(\Theta\left(\vec{q}, \overrightarrow{d_j}\right)\right)$$

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

Retrieval Models: Vector Space Model

Purdue University, Department of Computer Science

- Vector representation

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

Similarity Score	D1	D2	D3
Query	0.59	0.99	0.70

Retrieval Models: Vector Space Model

Purdue University, Department of Computer Science

Vector Coefficients

- The coefficients (vector elements) represent term evidence/ term importance
- 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:

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

First triple represents query term; second for document term

Retrieval Models: Vector Space Model

- Common vector weight components:
- lnc.ltc: widely used term weight
  - “l”: log(tf)+1
    - 0 if tf=0
  - “n”: no weight/normalization
  - “t”: log(N/df)
  - “c”: cosine normalization

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

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 Summary

- 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
- Generates 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

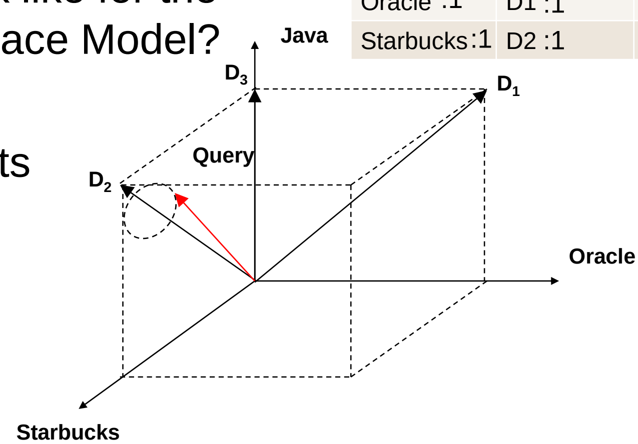
Disadvantages:

- 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

Back to Inverted Indexes

- What does an Inverted Index look like for the Vector Space Model?
- Words
- Documents
- *Weights*

Word	Doc	Doc	…
Java :1/3	D1 :1	D2 :1	D3 :1
Oracle :1	D1 :1		
Starbucks :1	D2 :1		



Retrieval Models: Vector Space Model

What is a good vector representation?

- 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"….

Retrieve Concepts, not Terms

- Problem: Query is necessarily an incomplete representation of information needed
  - Terms known to querier
  - Exact information presumably unknown
- Idea: Retrieve similar concepts, not similar terms
- Challenge: What is the space of concepts?
  - How do we map document to concept?
  - How does user specify concept?

52