

## **SNS COLLEGE OF ENGINEERING** Kurumbapalayam (Po), Coimbatore – 641 107

### **An Autonomous Institution**

Accredited by NBA – AICTE and Accredited by NAAC – UGC with 'A' Grade Approved by AICTE, New Delhi & Affiliated to Anna University, Chennai

## **DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

# **COURSE NAME : 19CS732 INFORMATION RETRIEVAL TECHNIQUES**

**IVYEAR / VII SEMESTER** 

**Unit 2- MODELING AND RETRIEVAL EVALUATION** 

**Topic 2 : Vector Space Model** 





# **Problem**

 $\succ$  How to determine important words in a document? ► Word sense?

 $\rightarrow$  Word *n*-grams (and phrases, idioms,...)  $\rightarrow$  terms  $\blacktriangleright$  How to determine the degree of importance of a term within a document and within the entire collection?

 $\succ$  How to determine the degree of similarity between a document and the query?  $\succ$  In the case of the web, what is the collection and what are the effects of links, formatting information, etc.?





### **Vector Space Model**

 $\succ$  The Vector Space Model (VSM) is a way of representing documents through the words that they contain ► It is a standard technique in Information Retrieval  $\succ$  The VSM allows decisions to be made about which documents are similar to each other and to keyword queries





## How it works: Overview

 $\succ$  Each document is broken down into a word frequency table  $\succ$  The tables are called vectors and can be stored as arrays  $\blacktriangleright$  A vocabulary is built from all the words in all documents in the system  $\succ$  Each document is represented as a vector based against the vocabulary





• Document A -"A dog and a cat."

а	dog	and	са
2	1	1	1

• Document B -"A frog."

а	frog	
1	1	

➤The vocabulary contains all words used  $\succ$ a, dog, and, cat, frog The vocabulary needs to be sorted  $\succ$ a, and, cat, dog, frog







Document A: "A dog and a cat."

Vector: (2,1,1,1,0) Document B: "A frog."

а	and	cat	dog	frog
2	1	1	1	0

а	and	cat	dog	frog
1	0	0	0	1

Vector: (1,0,0,0,1)

• Queries can be represented as vectors in the same way as documents:

$$-Dog = (0,0,0,1,0)$$

$$-Frog = (0,0,0,0,1)$$

-Dog and frog = ( 0,0,0,1,1)





Define:  $\bullet$ 

 $\Box$  wij > 0 whenever ki  $\in$  dj

 $\Box$  wiq >= 0 associated with the pair (ki,q)

 $\Box$  vec(dj) = (w1j, w2j, ..., wtj) V w2q, ..., wtq)

□To each term *ki*, associate a unit vector *vec(i)* 

 $\Box$  The *t* unit vectors, *vec(1), ..., vec(t)* form an *orthonormal basis* (embodying independence assumption) for the t-dimensional space for representing queries and documents



$$vec(q) = (w1q,$$



- How to compute the weights *wij* and *wiq*? **Q**quantification of intra-document content (similarity/semantic emphasis)
  - •*tf* factor, the *term frequency* within a document
  - **Quantification of inter-document separation (dis**similarity/significant discriminant)
  - *idf* factor, the *inverse* document frequency  $\Box wij = tf(i,j) * idf(i)$





*N* be the total number of docs in the collection *ni* be the number of docs which contain *ki freq(i,j)* raw frequency of *ki* within *dj* A normalized *tf* factor is given by f(i,j) = freq(i,j) / max(freq(l,j))where the maximum is computed over all terms which occur within the document dj The *idf* factor is computed as idf(i) = log(N/ni)

the *log* makes the values of *tf* and *idf* comparable.





Represent documents and queries as Vectors of term-based features Features: tied to occurrence of terms in collection  $\vec{d}_i = (t_{1,i}, t_{2,i}, \dots, t_{N,i}); \vec{q}_k = (t_{1,k}, t_{2,k}, \dots, t_{N,k})$ E.g. Solution 1: Binary features: t=1 if presense, 0 otherwise Similiarity: number of terms in common Dot product  $sim(\vec{q}_k, \vec{d}_j) = \sum^N t_{i,k} t_{i,j}$ 

> Unit-2/Modeling and Retrieval Evaluation /19CS732 Information Retrieval Techniques /Jebakumar Immanuel D/CSE/SNSCE





- Problem: Not all terms equally interesting
  - -E.g. the vs dog vs Levow

 $\vec{d}_i = (w_{1,i}, w_{2,i}, \dots, w_{N,i}); \vec{q}_k = (w_{1,k}, w_{2,k}, \dots, w_{N,k})$ 

- Solution: Replace binary term features with weights
  - –Document collection: term-by-document matrix
  - -View as vector in multidimensional space
    - Nearby vectors are related
  - –Normalize for vector length





Similarity = Dot product

$$sim(\vec{q}_k, \vec{d}_j) = \vec{q}_k \bullet \vec{d}_j = \sum_{i=1}^N w_{i,k} w_{i,j}$$

Normalization:

Normalize weights in advance

Normalize post-hoc

$$sim(\vec{q}_{k}, \vec{d}_{j}) = \frac{\sum_{i=1}^{N} w_{i,k} w_{i,j}}{\sqrt{\sum_{i=1}^{N} w_{i,k}^{2} \sqrt{\sum_{i=1}^{N} w_{i,j}^{2}}}}$$





# Activity

**Unit-2/Modeling and Retrieval Evaluation /19CS732 Information Retrieval** Techniques /Jebakumar Immanuel D/CSE/SNSCE





# Disadvantages

> assumes independence of index terms; not clear that this is bad though





# Advantages

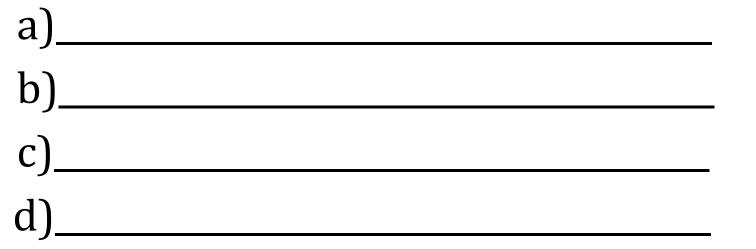
 $\succ$  term-weighting improves answer set quality  $\succ$  partial matching allows retrieval of docs that approximate the query conditions Cosine ranking formula sorts documents according to degree of similarity to the query



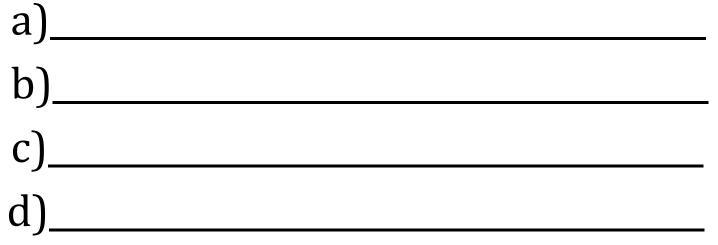


# Assessment 1

1. List out the Advantages of Vector Space Model



2. Identify the disadvantages of Vector Space Model









### **TEXT BOOKS:**

1. Ricardo Baeza-Yates and Berthier Ribeiro-Neto, —Modern Information Retrieval: The Concepts and Technology behind Search, Second Edition, ACM Press Books, 2011. 2. Ricci, F, Rokach, L. Shapira, B.Kantor, —Recommender Systems Handbook||, First Edition, 2011.

### **REFERENCES:**

1. C. Manning, P. Raghavan, and H. Schütze, —Introduction to Information Retrieval, Cambridge University Press, 2008.

2. Stefan Buettcher, Charles L. A. Clarke and Gordon V. Cormack, —Information Retrieval:

Implementing and Evaluating Search Engines, The MIT Press, 2010.

# **THANK YOU**

