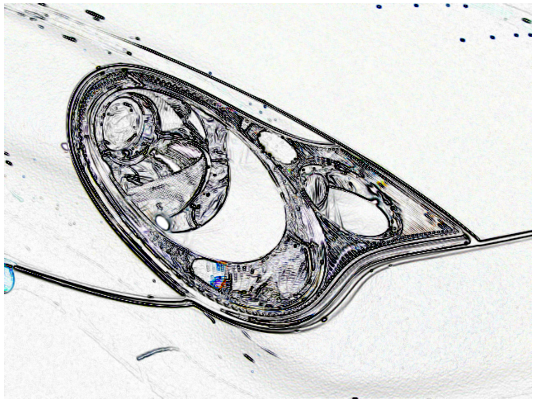
CSE578: Computer Vision

Spring 2016:

Image Indexing and Retrieval


Anoop M. Namboodiri

Center for Visual Information Technology

IIIT Hyderabad, INDIA

[Slides Generously Borrowed from Various Sources]

Content Based Image Retrieval

- Multimedia data is growing exponentially.
  - Cheap high quality digital imaging devices

  - Sharing of multimedia data on the internet

flickr BETA™   Picasa

- Content based organization and retrieval is a viable way of accessing this data.

Why CBIR?

- Historical Achieve
- Forensic documents
- Fingerprint & DNA matching
- Copyright search
- Security usage

Overview

- CBVR has two Approaches:
  - Attribute based
  - Object based
- CBVR can be done by:
  - Color
  - Texture
  - Shape
  - Spatial relationship
  - Semantic primitives
  - Objective Attribute
  - Subjective Attribute
  - Motion
  - Text & domain concepts

Overview

- CBVR has two phases:
  - Database Population phase
    - Video shot boundary detection
    - Key Frames selection
    - Feature extraction
  - Video Retrieval phase
    - Similarity measure

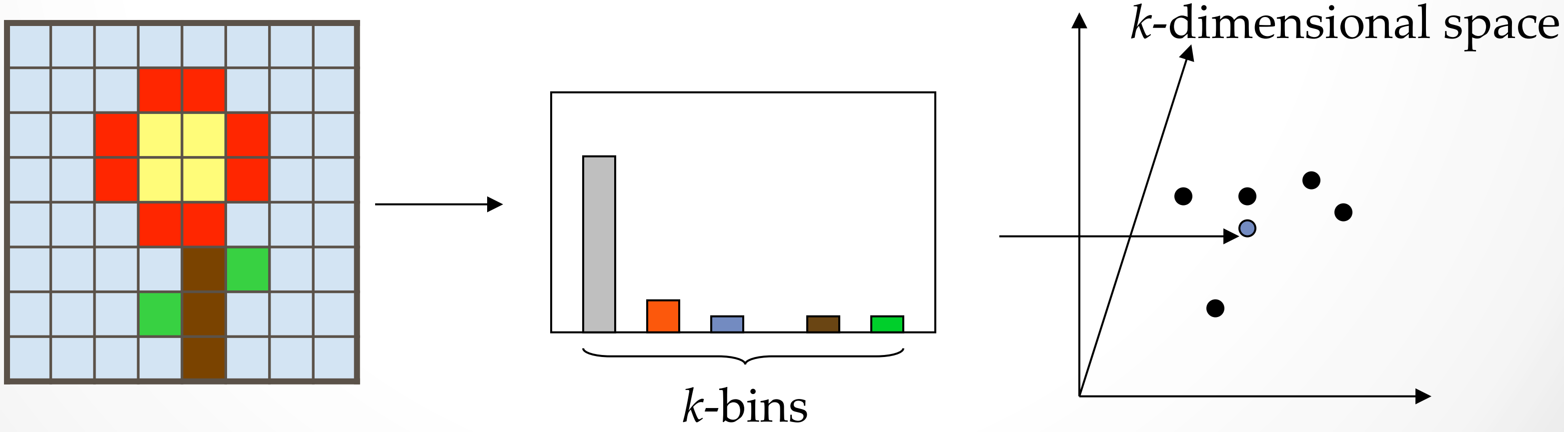
What do users look for?

- *In CBIR systems users look for ‘things’ not ‘stuff’.
  - More than global image properties
  - Traditional object recognition will not work
  - Two choices:
    - Rely on text to identify objects
    - Look at regions (objects or parts of objects)

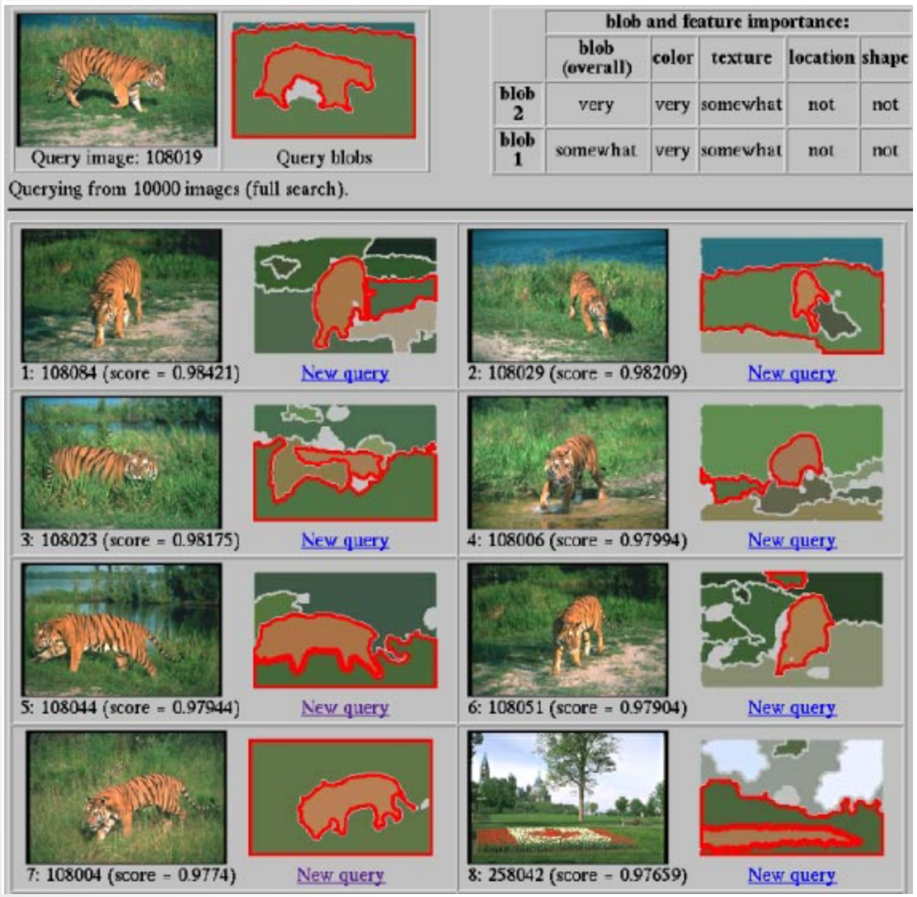
*Chad Carson: Blobworld

Initial Attempts

- Images represented using simple features
  - Color Histograms
  - Color moments
  - Texture descriptors
- A distance metric (e.g., Euclidean) is used to measure similarity



Initial Successful Systems



Blob World: Carson, Belongie, Malik

SIMPLIcity: Wnag, Li, Wiederhold

Practical CBIR systems

- Practical large scale deployment of CBIR systems require.
  - Efficient indexing and retrieval of thousands of documents.
  - Flexible framework for retrieval based on various types of features. For example Specialized features for highly specific sub domains like faces, vehicles, monuments.
  - Ability to scale up to millions of images without a significant performance trade off.

Lessons From Text Retrieval

- Large scale text retrieval systems have been successfully deployed.
  - Search Engines like

Google YAHOO! bing

- Efficient indexing and retrieval of millions of documents has been achieved.
- The text retrieval frameworks are adaptive enough to be applied to specialized domains.

Froogle BETA CiteSeer Scientific Literature Digital Library

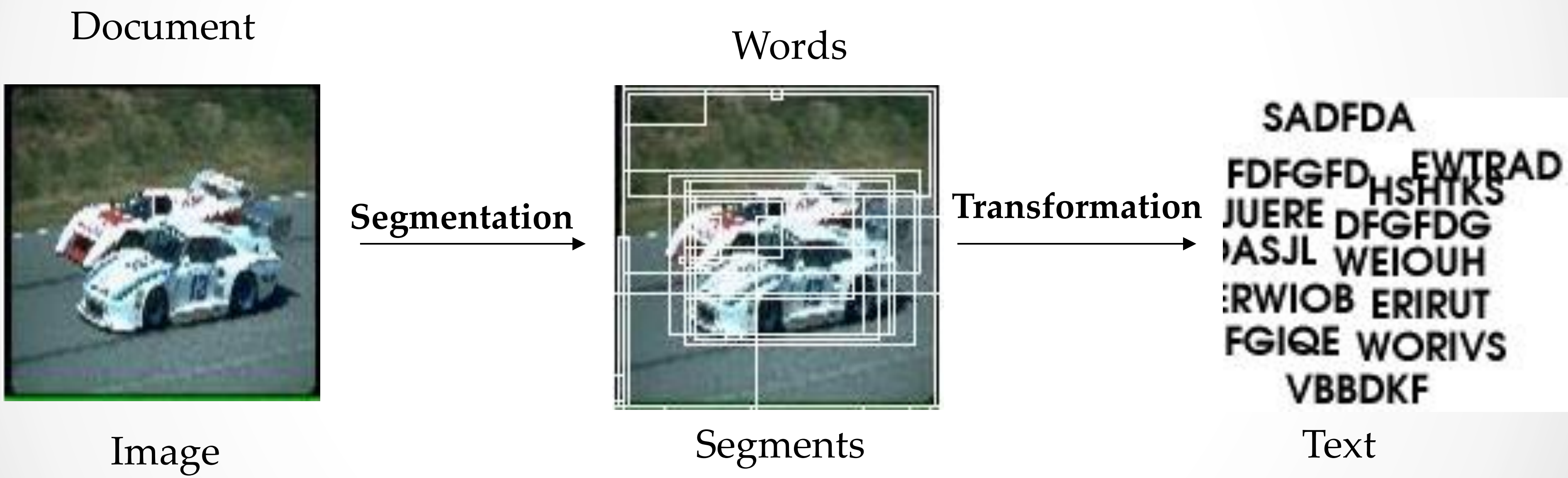
Froogle Example

Query: Flared Cashmere Sweater Dress

Looking For: Flared Grey Long Sleeve Dress

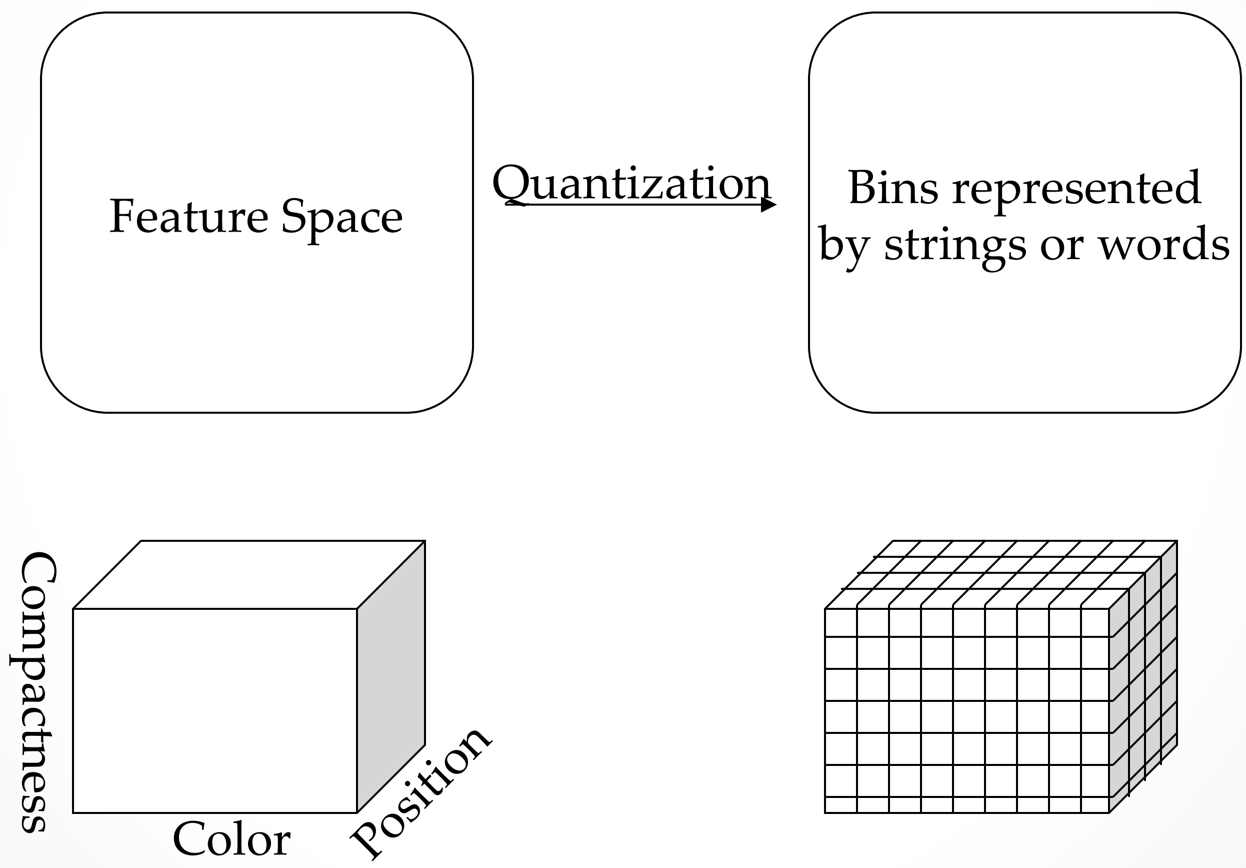
Virtual Textual Representation

- Images as text documents.



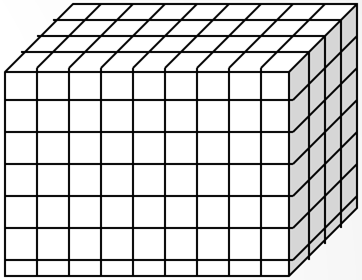
- Color (YUV), compactness and location of segment are used to encode the segment as text.

Transformation

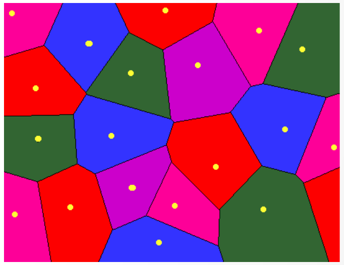


Transformation

- Quantization can be achieved in a number of ways.
  - Uniform vector space quantization for data set with a uniform feature point distribution.

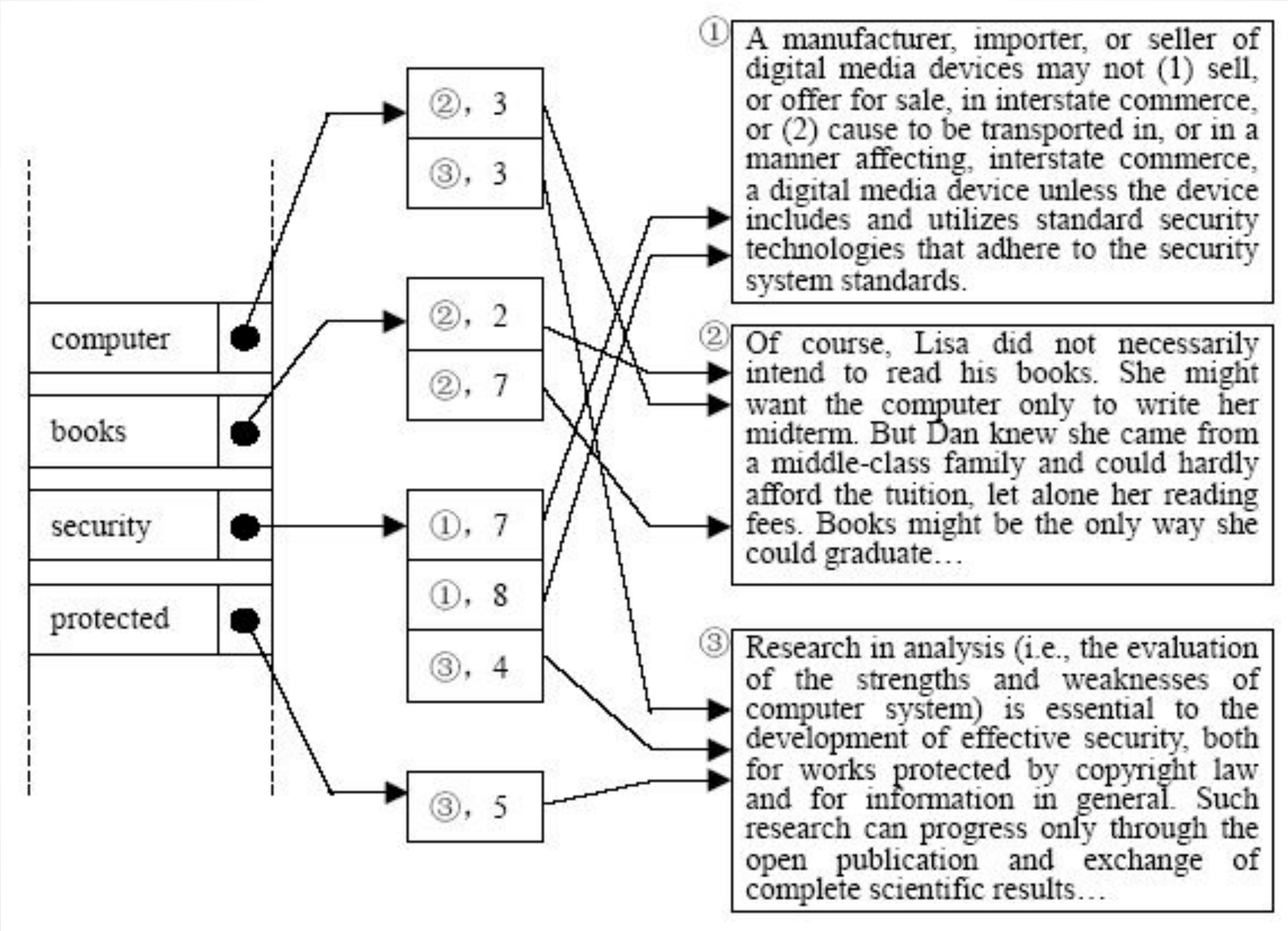


  - Density based quantization of the feature space can be achieved with simple k-means quantization.



- Irrespective of the quantization applied each cell in the vector space has a representative string.
- Each image segment is assigned to a cell and is assigned the representative string of the cell.

Inverted Index

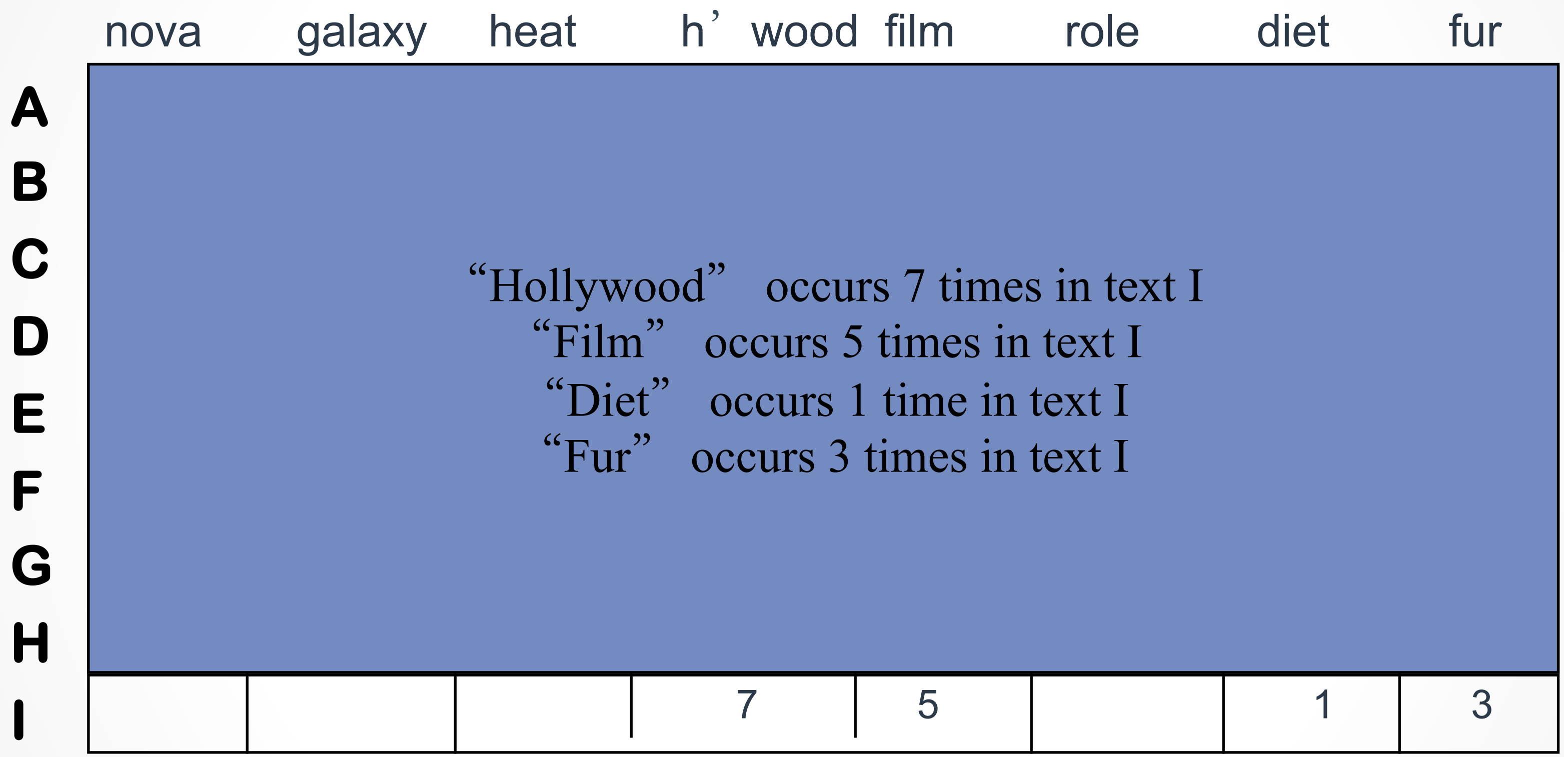


Document as a Word Histogram

	nova	galaxy	heat	h’wood	film	role	diet	fur
A	10	5	3					
B								
C								
D								
E								
F								
G								
H								
I								

“Nova” occurs 10 times in text A
“Galaxy” occurs 5 times in text A
“Heat” occurs 3 times in text A
(Blank means 0 occurrences.)

Document as a Word Histogram



Documents as Word Histograms

	nova	galaxy	heat	h’wood	film	role	diet	fur
A	10	5	3					
B	5	10						
C				10	8	7		
D				9	10	5		
E							10	10
F							9	10
G	5	7			9			
H		6	10	2	8			
I				7	5		1	3

Assigning Weights to Terms

- Binary Weights
- Raw term frequency
- tf x idf
    - Recall the Zipf distribution
    - Want to weight terms highly if they are
        - frequent in relevant documents … BUT
        - infrequent in the collection as a whole

Binary Weights

- Only the presence (1) or absence (0) of a term is included in the vector

docs	*t1*	*t2*	*t3*
D1	1	0	1
D2	1	0	0
D3	0	1	1
D4	1	0	0
D5	1	1	1
D6	1	1	0
D7	0	1	0
D8	0	1	0
D9	0	0	1
D10	0	1	1
D11	1	0	1

Raw Term Weights

- The frequency of occurrence for the term in each document is included in the vector

docs	*t1*	*t2*	*t3*
D1	2	0	3
D2	1	0	0
D3	0	4	7
D4	3	0	0
D5	1	6	3
D6	3	5	0
D7	0	8	0
D8	0	10	0
D9	0	0	1
D10	0	3	5
D11	4	0	1

TF-IDF Measure

- Term frequency (tf)
- Inverse document frequency (idf)
  - A way to deal with the problems of the Zipf distribution
- Goal: assign a tf * idf weight to each term in each document

Computing TF*IDF

$$w_{ik} = tf_{ik} * \log(N / n_k)$$

T_k = term k

tf_{ik} = frequency of term T_k in document D_i

idf_k = inverse document frequency of term T_k in C

N = total number of documents in the collection C

n_k = the number of documents in C that contain T_k

$$idf_k = \log\left(\frac{N}{n_k}\right)$$

Why IDF?

- IDF provides high values for rare words and low values for common words:

For a collection of 10000 documents:

$$\log\left(\frac{10000}{10000}\right) = 0$$

$$\log\left(\frac{10000}{5000}\right) = 0.301$$

$$\log\left(\frac{10000}{20}\right) = 2.698$$

$$\log\left(\frac{10000}{1}\right) = 4$$

- Useful for Ranking

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.

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