

CarnegieMellon

15-826: Multimedia Databases and Data Mining

Lecture #14: Text - part III:
Vector space model and clustering

C. Faloutsos

CarnegieMellon

Must-Read Material

- MM Textbook, Chapter 6

Outline

Goal: 'Find similar / interesting things'

- Intro to DB
- ➔ • Indexing - similarity search
- Data Mining

Indexing - Detailed outline

- primary key indexing
- secondary key / multi-key indexing
- spatial access methods
- fractals
- ➔ • text
- multimedia
- ...

CarnegieMellon

Text - Detailed outline



- text
 - problem
 - full text scanning
 - inversion
 - signature files
 - ➔ – clustering
 - information filtering and LSI

15-826 Copyright (c) 2019 C. Faloutsos 5

CarnegieMellon


Problem

- How to find doc's with “data mining”?




15-826 Copyright (c) 2019 C. Faloutsos 6

CarnegieMellon



Conclusion

- How to find doc's with "data mining"?
- ✓ • A1: full text scanning
 - A1.1: string editing distance
- ✓ • A2: inversion
 - Elias Codes
- ✓ • (A3: signature files – 'Bloom filters')
- A4: vector space model + clustering



15-826 Copyright (c) 2019 C. Faloutsos 7

CarnegieMellon

Vector Space Model and Clustering

- keyword queries (vs Boolean)
- each document: -> vector (HOW?)
- each query: -> vector
- search for 'similar' vectors

15-826 Copyright (c) 2019 C. Faloutsos 8

CarnegieMellon

Vector Space Model and Clustering

- main idea:

document

...data...

'indexing'

aaron data zoo

V (= vocabulary size)

15-826 Copyright (c) 2019 C. Faloutsos 9

CarnegieMellon

Vector Space Model and Clustering

Then, group nearby vectors together

- Q1: cluster search?
- Q2: cluster generation?

Two significant contributions

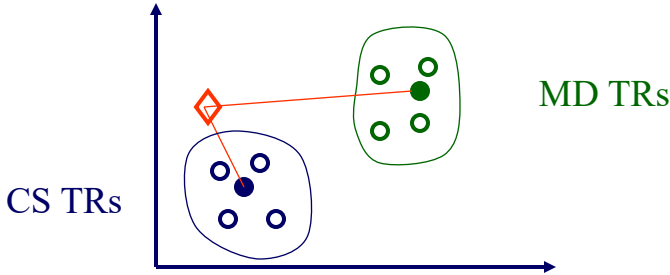
- ranked output
- relevance feedback

15-826 Copyright (c) 2019 C. Faloutsos 10

CarnegieMellon

Vector Space Model and Clustering

- cluster search: visit the (k) closest superclusters; continue recursively



CS TRs

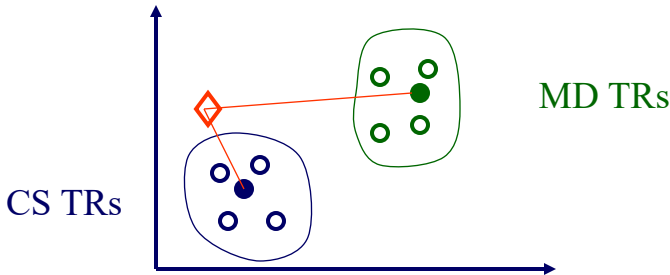
MD TRs

15-826 Copyright (c) 2019 C. Faloutsos 11

CarnegieMellon

Vector Space Model and Clustering

- ranked output: easy!



CS TRs

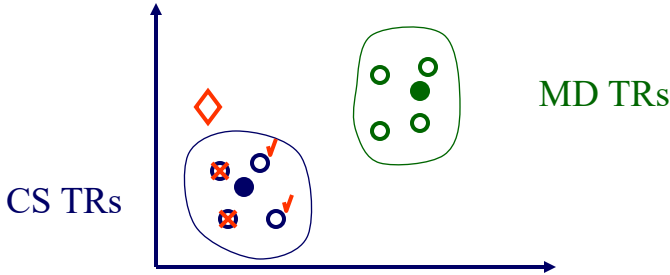
MD TRs

15-826 Copyright (c) 2019 C. Faloutsos 12

CarnegieMellon

Vector Space Model and Clustering

- relevance feedback (brilliant idea) [Rocchio '73]



CS TRs

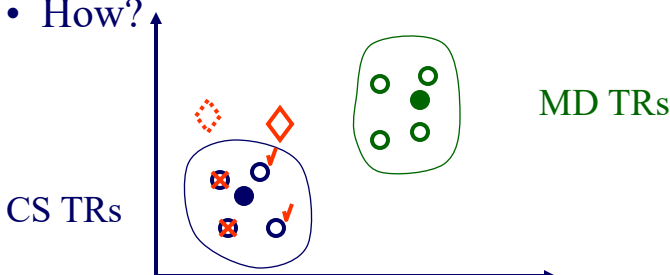
MD TRs

15-826 Copyright (c) 2019 C. Faloutsos 13

CarnegieMellon

Vector Space Model and Clustering

- relevance feedback (brilliant idea) [Rocchio '73]
- How?



CS TRs

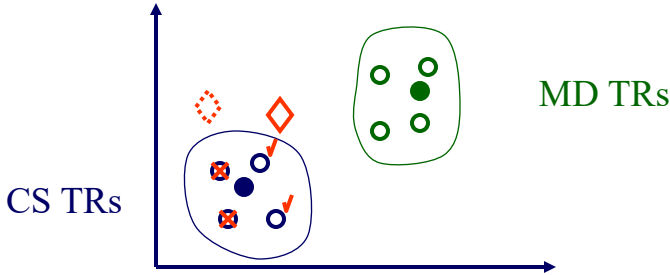
MD TRs

15-826 Copyright (c) 2019 C. Faloutsos 14

CarnegieMellon

Vector Space Model and Clustering

- How? A: by adding the ‘good’ vectors and subtracting the ‘bad’ ones



CS TRs

MD TRs

15-826 Copyright (c) 2019 C. Faloutsos 15

CarnegieMellon

Outline - detailed

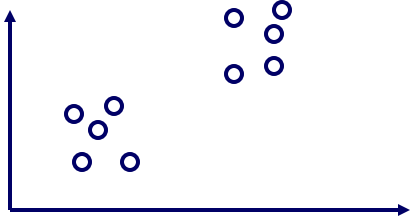
- main idea
- cluster search
- ➔ • cluster generation
- evaluation

15-826 Copyright (c) 2019 C. Faloutsos 16

CarnegieMellon

Cluster generation

- Problem:
 - given N points in V dimensions,
 - group them

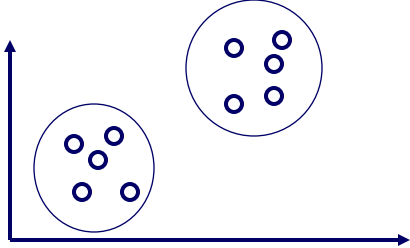


15-826 Copyright (c) 2019 C. Faloutsos 17

CarnegieMellon

Cluster generation

- Problem:
 - given N points in V dimensions,
 - group them



15-826 Copyright (c) 2019 C. Faloutsos 18

Cluster generation

We need

- Q1: document-to-document similarity
- Q2: document-to-cluster similarity

Cluster generation

Q1: document-to-document similarity
(recall: 'bag of words' representation)

- D1: { 'data', 'retrieval', 'system' }
- D2: { 'lung', 'pulmonary', 'system' }
- distance/similarity functions?

CarnegieMellon

Cluster generation

A1: # of words in common

A2: normalized by the vocabulary sizes

A3: etc

About the same performance - prevailing one:
cosine similarity

15-826

Copyright (c) 2019 C. Faloutsos

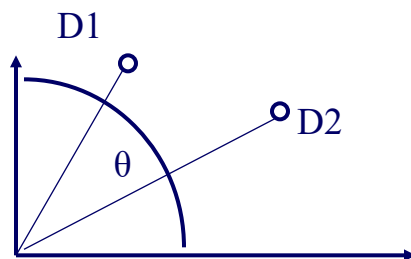
21

CarnegieMellon

Cluster generation

cosine similarity:

$$\text{similarity}(D1, D2) = \cos(\theta) = \frac{\text{sum}(v_{1,i} * v_{2,i})}{\text{len}(v_1) * \text{len}(v_2)}$$



15-826

Copyright (c) 2019 C. Faloutsos

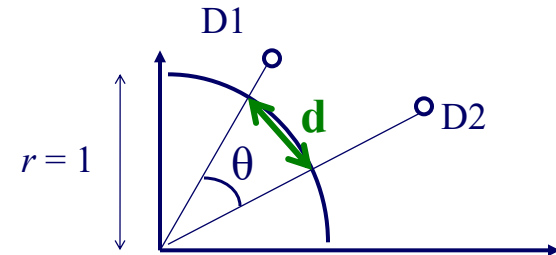
22

CarnegieMellon

Cluster generation

cosine similarity - observations:

- related to the **Euclidean distance**
- weights $v_{i,j}$: according to tf/idf



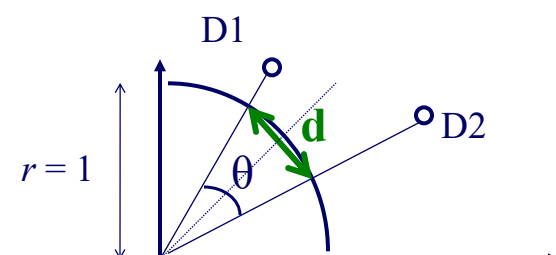
15-826 Copyright (c) 2019 C. Faloutsos 23

CarnegieMellon

Cluster generation

cosine similarity - observations:

- related to the **Euclidean distance**
- weights $v_{i,j}$: according to tf/idf



$$d = 2 * \sin(\theta/2)$$

$$d^2 = 2 * (1 - \cos(\theta))$$

15-826 Copyright (c) 2019 C. Faloutsos 24

CarnegieMellon

Cluster generation

tf ('term frequency')

high, if the term appears very often in this document.

idf ('inverse document frequency')

penalizes 'common' words, that appear in almost every document

15-826

Copyright (c) 2019 C. Faloutsos

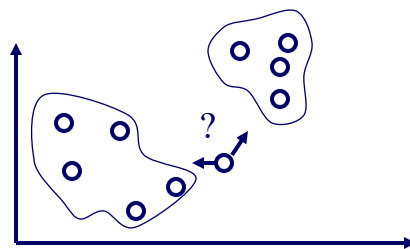
25

CarnegieMellon

Cluster generation

We need

- Q1: document-to-document similarity
- • Q2: document-to-cluster similarity



15-826

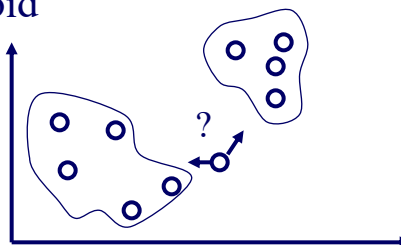
Copyright (c) 2019 C. Faloutsos

26

CarnegieMellon

Cluster generation

- A1: min distance ('single-link')
- A2: max distance ('all-link')
- A3: avg distance
- A4: distance to centroid



15-826

Copyright (c) 2019 C. Faloutsos

27

CarnegieMellon

Cluster generation

- A1: min distance ('single-link')
 - leads to elongated clusters
- A2: max distance ('all-link')
 - many, small, tight clusters
- A3: avg distance
 - in between the above
- A4: distance to centroid
 - fast to compute

15-826

Copyright (c) 2019 C. Faloutsos

28

CarnegieMellon

Cluster generation

We have

- document-to-document similarity
- document-to-cluster similarity

Q: How to group documents into ‘natural’ clusters

15-826

Copyright (c) 2019 C. Faloutsos

29

CarnegieMellon

Cluster generation

A: *many-many* algorithms - in two groups
[VanRijsbergen]:

- theoretically sound ($O(N^2)$)
 - independent of the insertion order
- iterative ($O(N)$, $O(N \log(N))$)

15-826

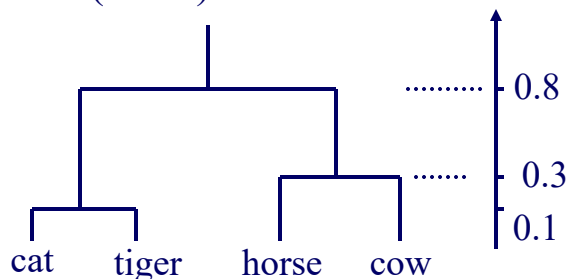
Copyright (c) 2019 C. Faloutsos

30

CarnegieMellon

Cluster generation - 'sound' methods

- Approach#1: dendrograms - create a hierarchy (bottom up or top-down) - choose a cut-off (how?) and cut



15-826

Copyright (c) 2019 C. Faloutsos

31

CarnegieMellon

Cluster generation - 'sound' methods

- Approach#2: min. some statistical criterion (eg., sum of squares from cluster centers)
 - like 'k-means'
 - but how to decide 'k' ?

15-826

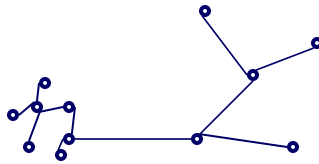
Copyright (c) 2019 C. Faloutsos

32

CarnegieMellon

Cluster generation - 'sound' methods

- Approach#3: Graph theoretic [Zahn]:
 - build MST;
 - delete edges longer than $3 \times$ std of the local average



15-826

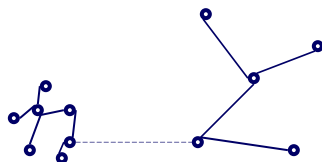
Copyright (c) 2019 C. Faloutsos

33

CarnegieMellon

Cluster generation - 'sound' methods

- Result:
 - why '3'?
 - variations
 - Complexity?



15-826

Copyright (c) 2019 C. Faloutsos

34

CarnegieMellon

Cluster generation - 'iterative' methods

general outline:

- Choose 'seeds' (how?)
- assign each vector to its closest seed (possibly adjusting cluster centroid)
- possibly, re-assign some vectors to improve clusters

Fast and practical, but 'unpredictable'

15-826

Copyright (c) 2019 C. Faloutsos

35

CarnegieMellon

Cluster generation - 'iterative' methods

general outline:

- Choose 'seeds' (how?)
- assign each vector to its closest seed (possibly adjusting cluster centroid)
- possibly, re-assign some vectors to improve clusters

Fast and practical, but 'unpredictable'

15-826

Copyright (c) 2019 C. Faloutsos

36

CarnegieMellon

Cluster generation

one way to estimate # of clusters k : the 'cover coefficient' [Can+] ~ SVD


15-826

Copyright (c) 2019 C. Faloutsos

37

CarnegieMellon

Outline - detailed

- main idea
- cluster search
- cluster generation
-  • evaluation

15-826

Copyright (c) 2019 C. Faloutsos

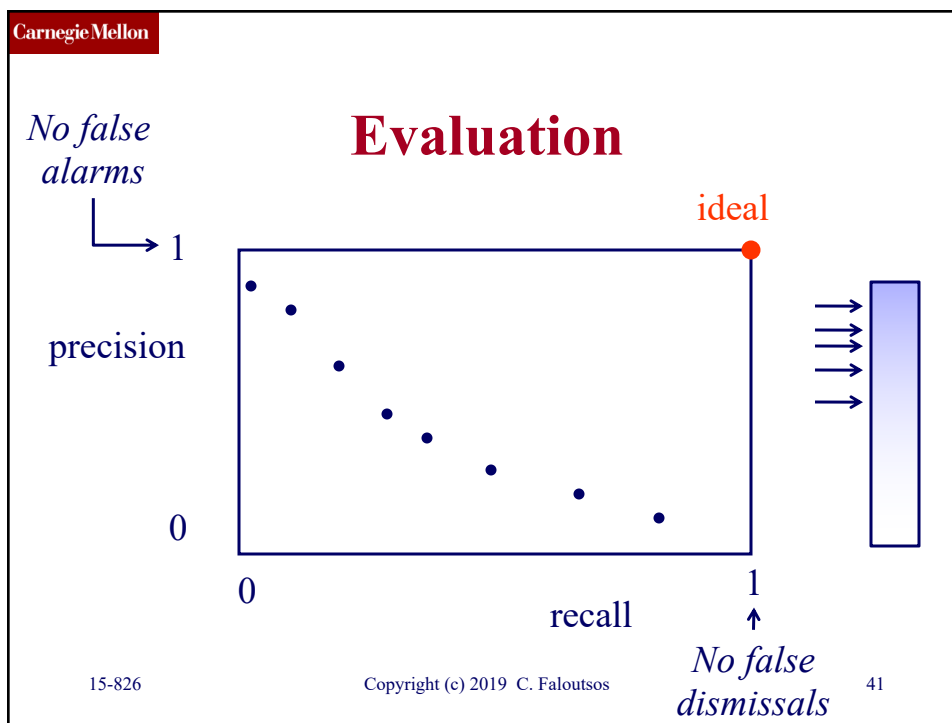
38

Evaluation

- Q: how to measure ‘goodness’ of one distance function vs another?
- A: ground truth (by humans) and
 - ‘precision’ and ‘recall’


Evaluation

- $\text{precision} = (\text{retrieved \& relevant}) / \text{retrieved}$
 - 100% precision -> no false alarms
- $\text{recall} = (\text{retrieved \& relevant}) / \text{relevant}$
 - 100% recall -> no false dismissals



CarnegieMellon

Evaluation

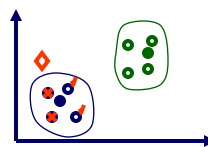


- compressing such a curve into a single number:
 - 11-point average precision
 - etc

15-826 Copyright (c) 2019 C. Faloutsos 42

Conclusions – main ideas

- ‘bag of words’ idea + keyword queries
- Cosine similarity
- Ranked output
- Relevance feedback



Conclusion



- How to find doc's with “data mining”?
- ✓ • A1: full text scanning
 - A1.1: string editing distance
- ✓ • A2: inversion
 - Elias Codes
- ✓ • (A3: signature files – ‘Bloom filters’)
- A4: vector space model + clustering



References

- *Modern Information Retrieval* [R. Baeza-Yates](#), [Acm Press](#), [Berthier Ribeiro-Neto](#), February 1999
- Can, F. and E. A. Ozkarahan (Dec. 1990). "Concepts and Effectiveness of the Cover-Coefficient-Based Clustering Methodology for Text Databases." *ACM TODS* 15(4): 483-517.
- Noreault, T., M. McGill, et al. (1983). A Performance Evaluation of Similarity Measures, Document Term Weighting Schemes and Representation in a Boolean Environment. *Information Retrieval Research*, Butterworths.

References

- Rocchio, J. J. (1971). *Relevance Feedback in Information Retrieval. The SMART Retrieval System - Experiments in Automatic Document Processing*. G. Salton. Englewood Cliffs, New Jersey, Prentice-Hall Inc.
- Salton, G. (1971). *The SMART Retrieval System - Experiments in Automatic Document Processing*. Englewood Cliffs, New Jersey, Prentice-Hall Inc.

References

- Salton, G. and M. J. McGill (1983). Introduction to Modern Information Retrieval, McGraw-Hill.
- Van-Rijsbergen, C. J. (1979). Information Retrieval. London, England, Butterworths.
- Zahn, C. T. (Jan. 1971). "Graph-Theoretical Methods for Detecting and Describing Gestalt Clusters." IEEE Trans. on Computers C-20(1): 68-86.