Text Clustering

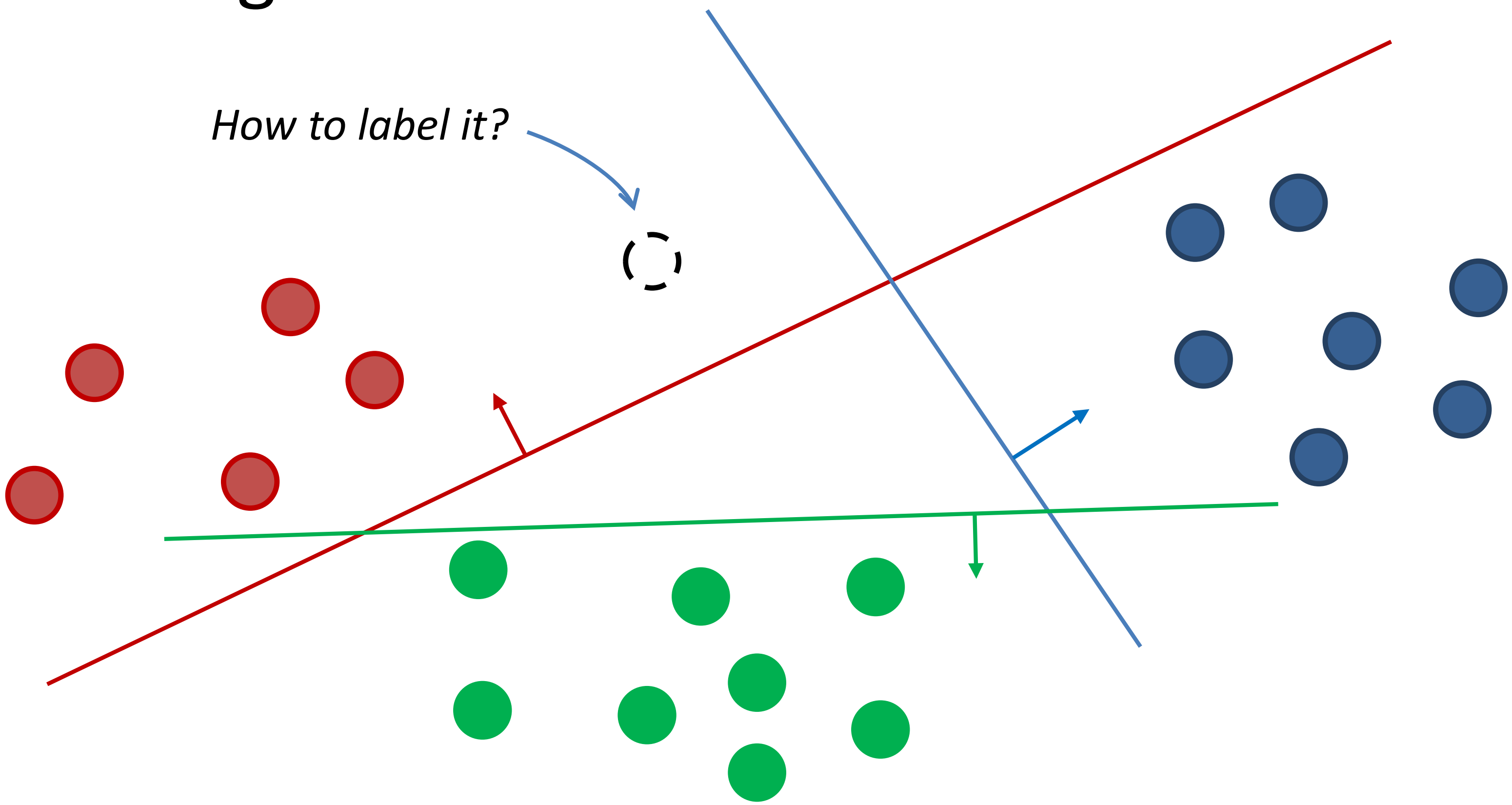
Hongning Wang
CS@UVa

Today's lecture

- Clustering of text documents
  - Problem overview
    - Applications
  - Distance metrics
  - Two basic categories of clustering algorithms
  - Evaluation metrics

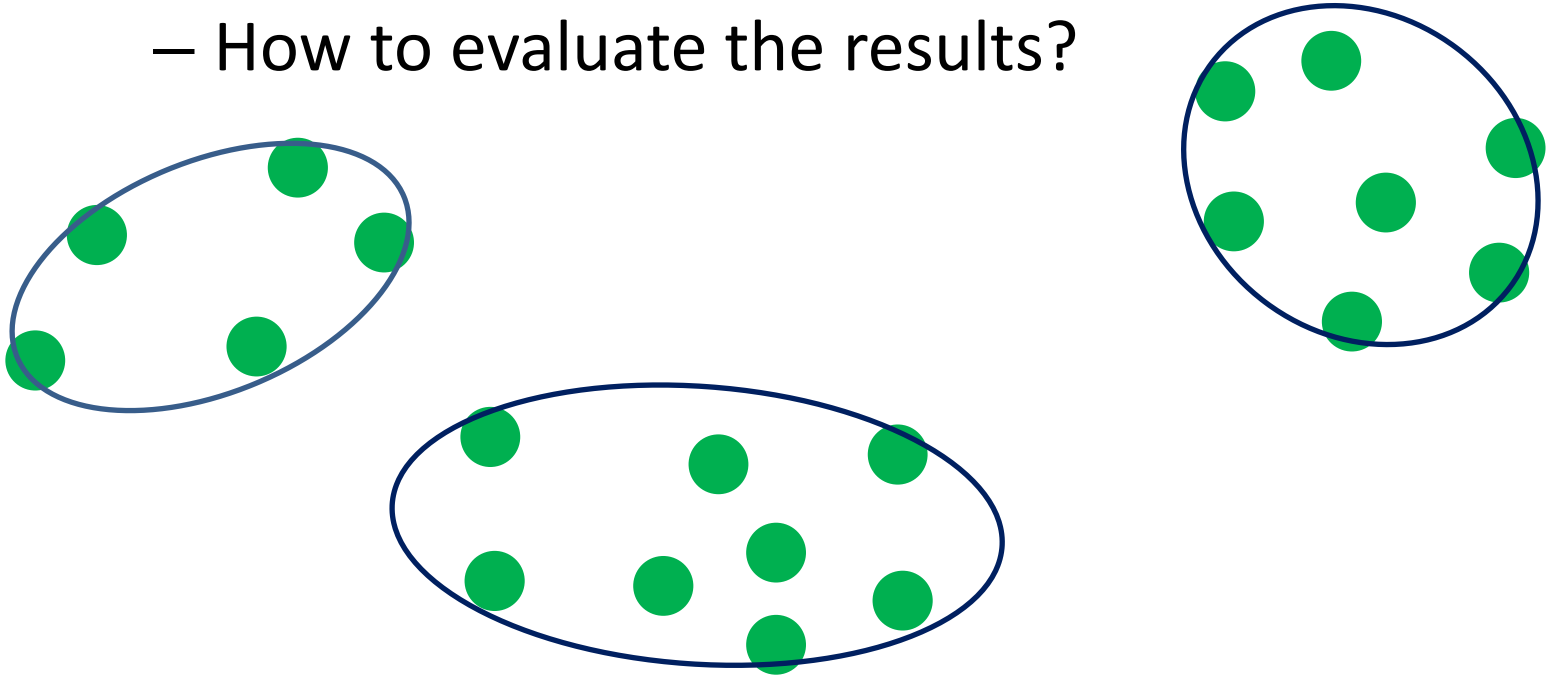
Clustering v.s. Classification

- Assigning documents to its corresponding categories



Clustering problem in general

- Discover “natural structure” of data
  - What is the criterion?
  - How to identify them?
  - How to evaluate the results?



Clustering problem in general

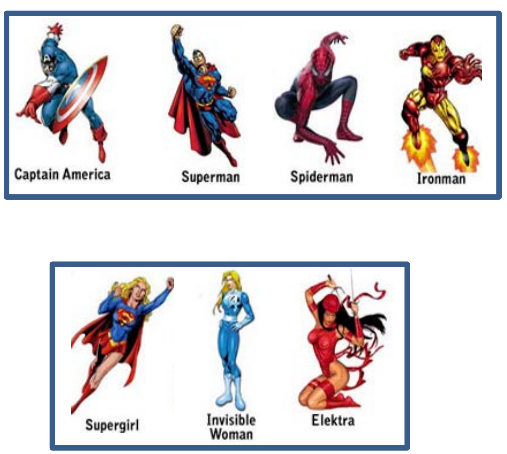
- Clustering - the process of grouping a set of objects into clusters of similar objects
  - Basic criteria
    - high intra-cluster similarity
    - low inter-cluster similarity
  - No (little) supervision signal about the underlying clustering structure
  - Need similarity/distance as guidance to form clusters

What is the “natural grouping”?

Clustering is very subjective!

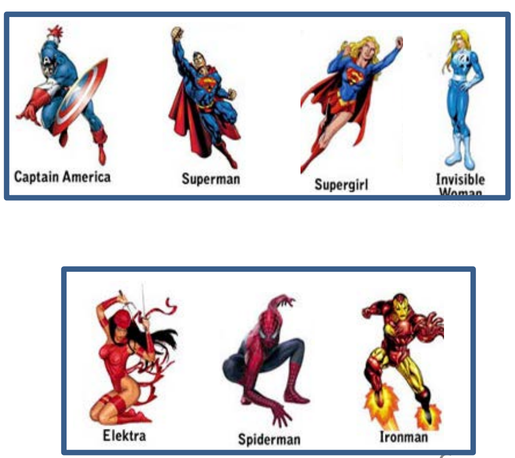
Distance metric is important!

group by gender

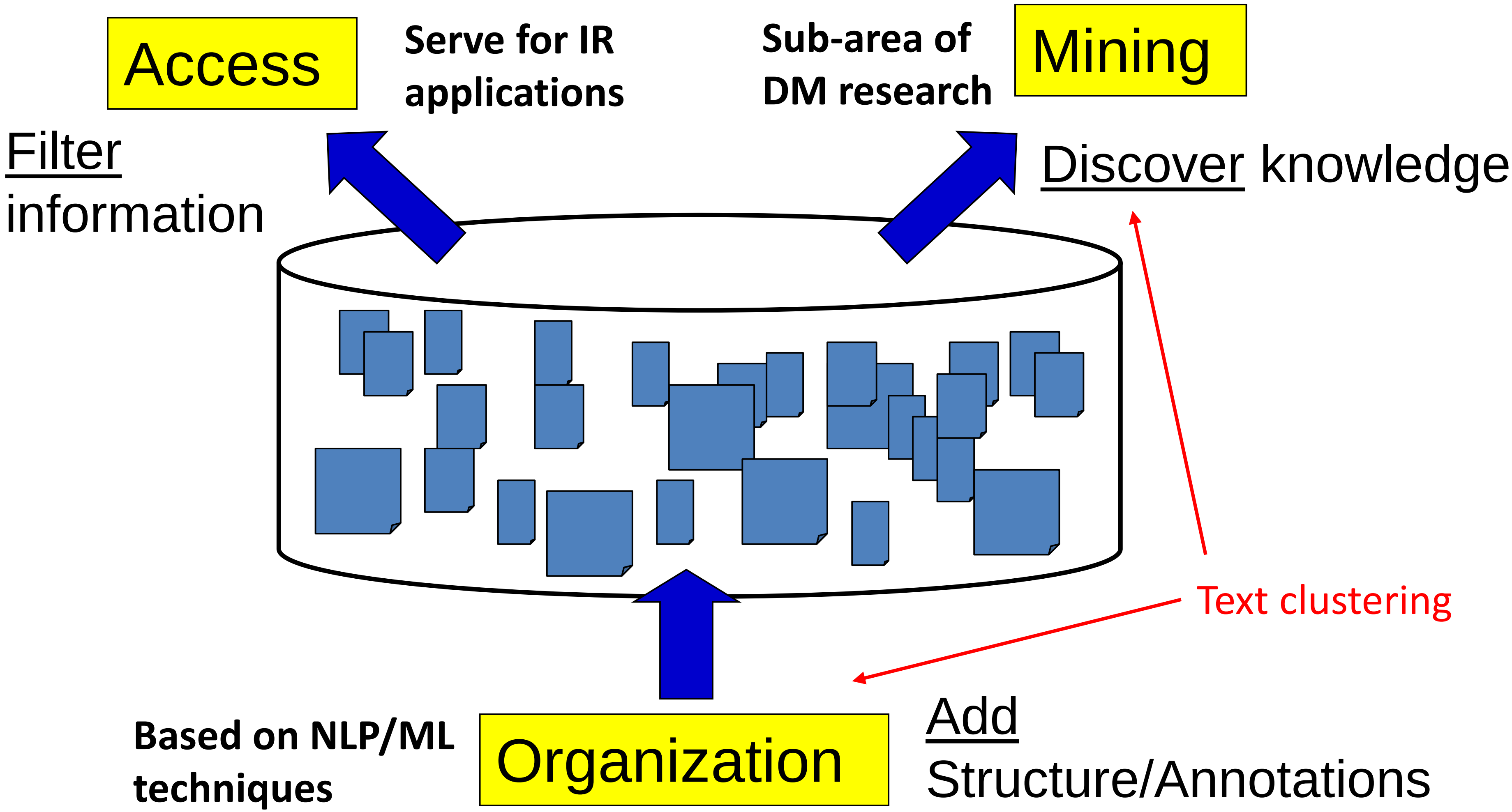


group by source of ability

group by costume

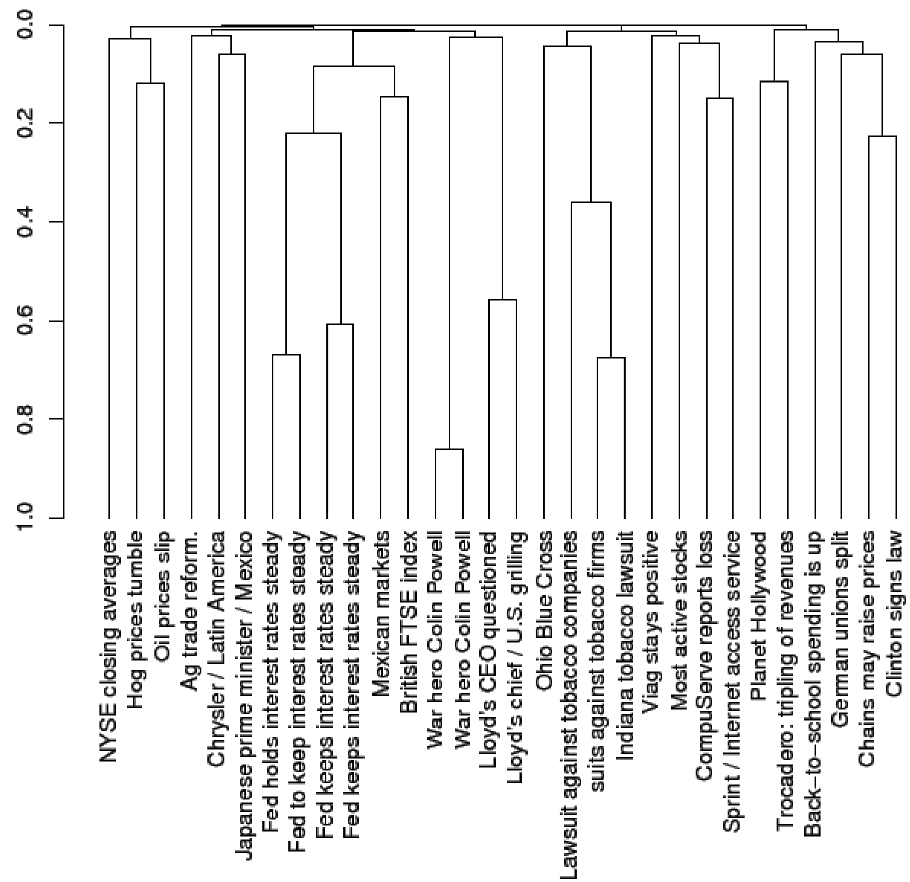


Clustering in text mining



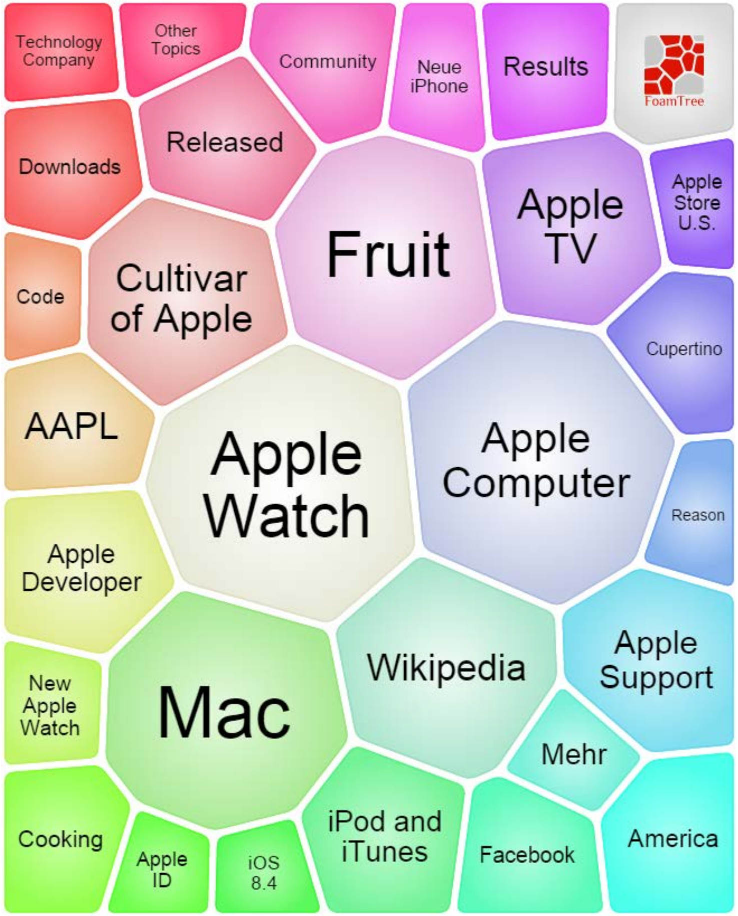
Applications of text clustering

- Organize document collections
  – Automatically identify hierarchical/topical relation among documents



Applications of text clustering

- Grouping search results
  - Organize documents by topics
  - Facilitate user browsing

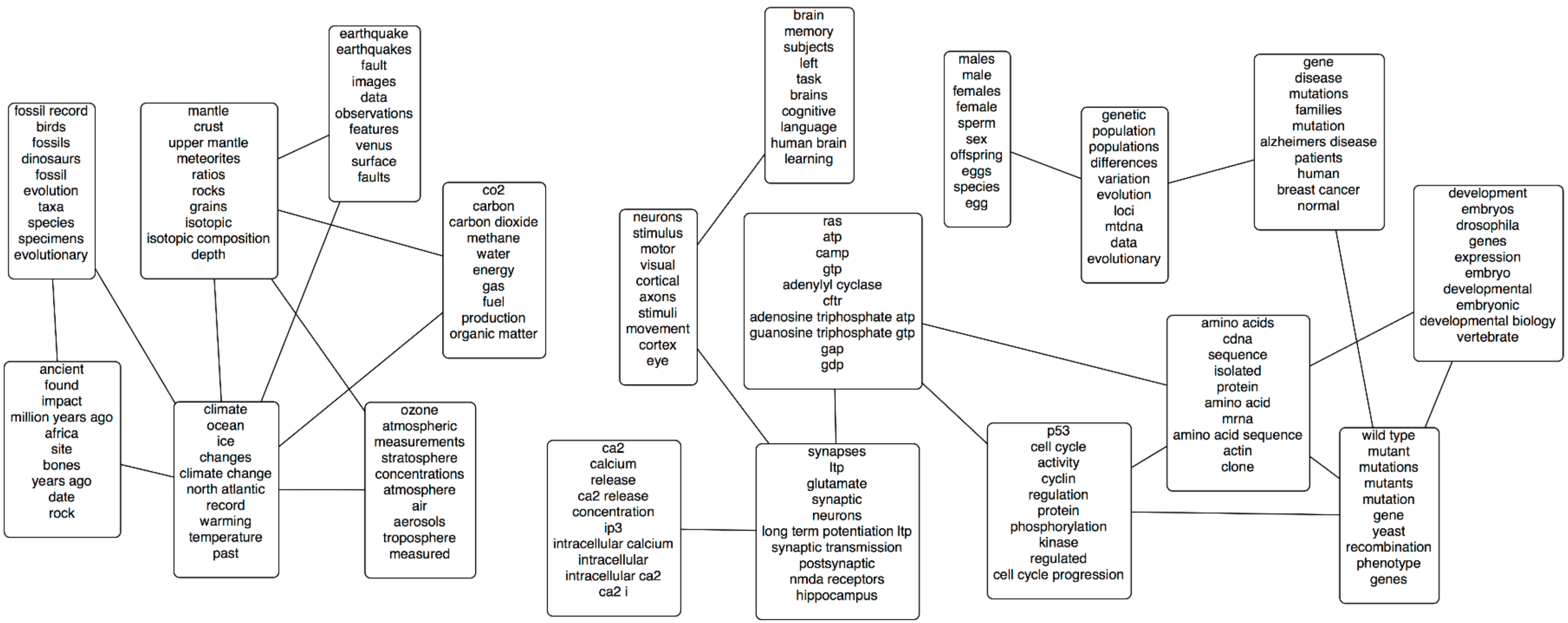


http://search.carrot2.org/stable/search

Applications of text clustering

- Topic modeling
  - Grouping words into topics

Will be discussed later separately



Distance metric

- Basic properties
  - Positive separation
    - $D(x, y) > 0, \forall x \neq y$
    - $D(x, y) = 0$, i.f.f., $x = y$
  - Symmetry
    - $D(x, y) = D(y, x)$
  - Triangle inequality
    - $D(x, y) \leq D(x, z) + D(z, y)$

Typical distance metric

- Minkowski metric
  - $d(x,y) = \sqrt[p]{\sum_{i=1}^{V}(x_i - y_i)^p}$
    - When $p = 2$, it is Euclidean distance
- Cosine metric
  - $d(x,y) = 1 - cosine(x,y)$
    - when $|x|^2 = |y|^2 = 1$, $1 - cosine(x,y) = \frac{r^2}{2}$

Typical distance metric

- Edit distance
  - Count the minimum number of operations required to transform one string into the other
    - Possible operations: insertion, deletion and replacement

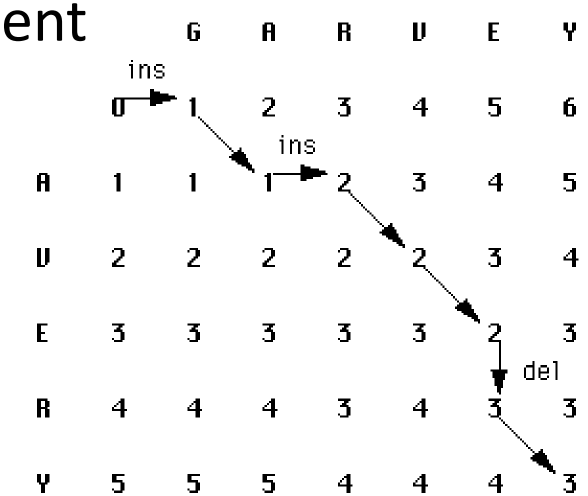


Figure 1. d(i,j) Matrix with Minimal Path Identified

Can be efficiently solved by dynamic programming

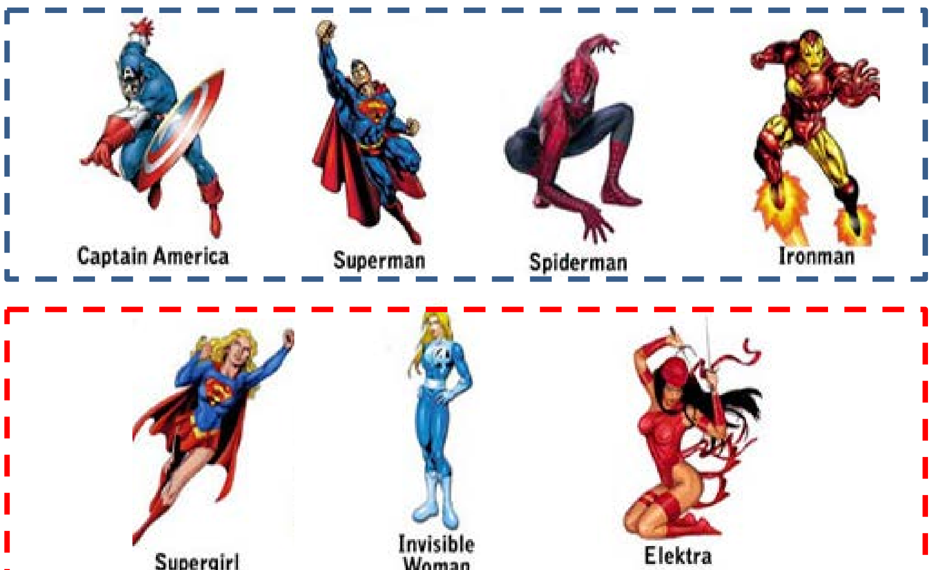
Typical distance metric

- Edit distance
  - Count the minimum number of operations required to transform one string into the other
    - Possible operations: insertion, deletion and replacement
  - Extent to distance between sentences
    - Word similarity as cost of replacement
      - "terrible" -> "bad": low cost
      - "terrible" -> "terrific": high cost

      Lexicon or distributional semantics
    - Preserving word order in distance computation

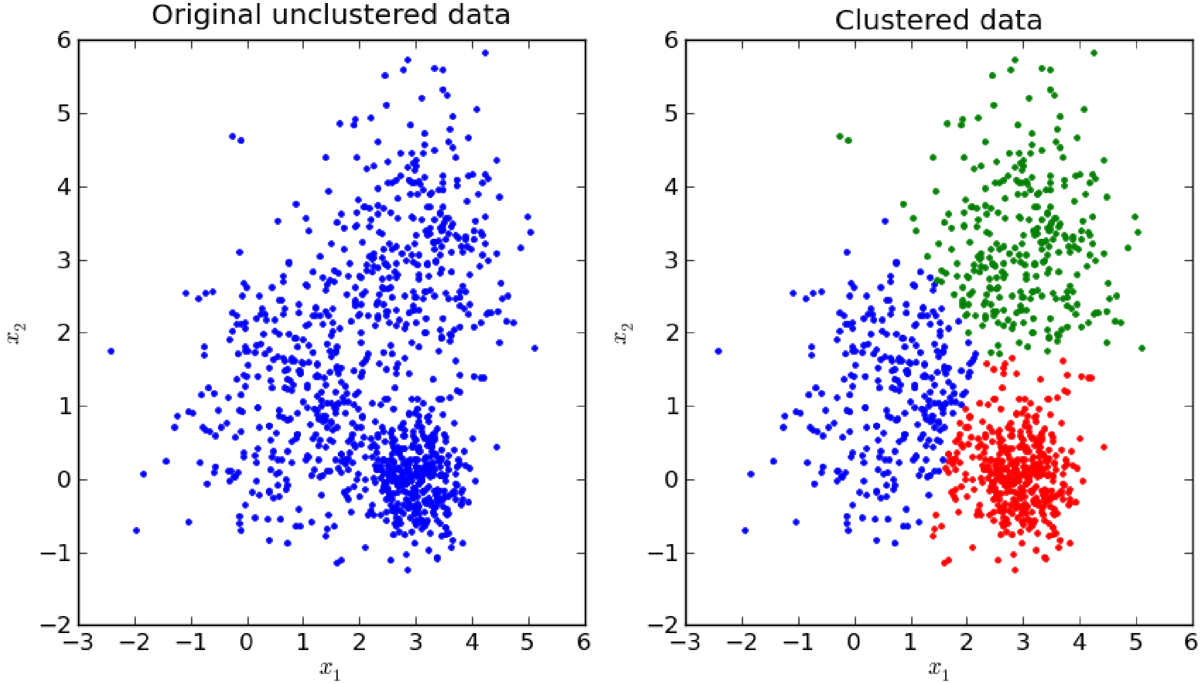
Clustering algorithms

- Partitional clustering algorithms
  - Partition the instances into different groups
  - Flat structure
    - Need to specify the number of classes in advance



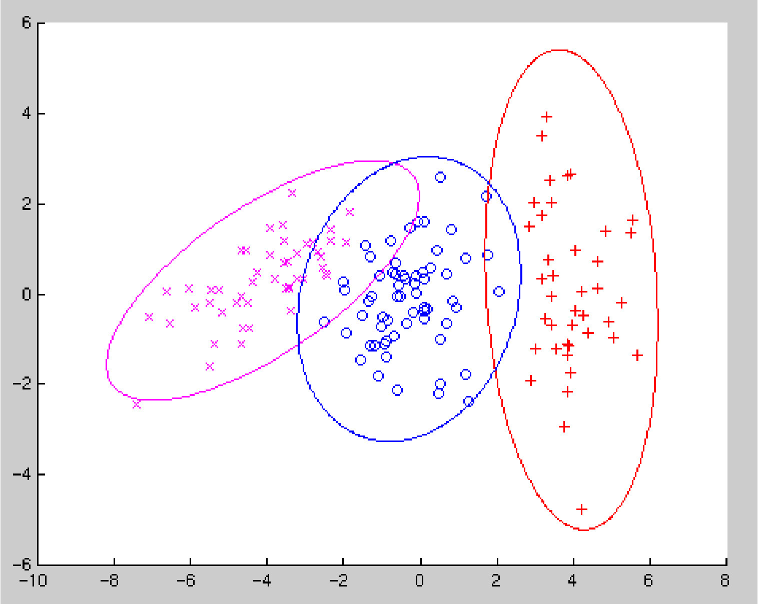
Clustering algorithms

- Typical partitional clustering algorithms
  - *k*-means clustering
    - Partition data by its closest mean



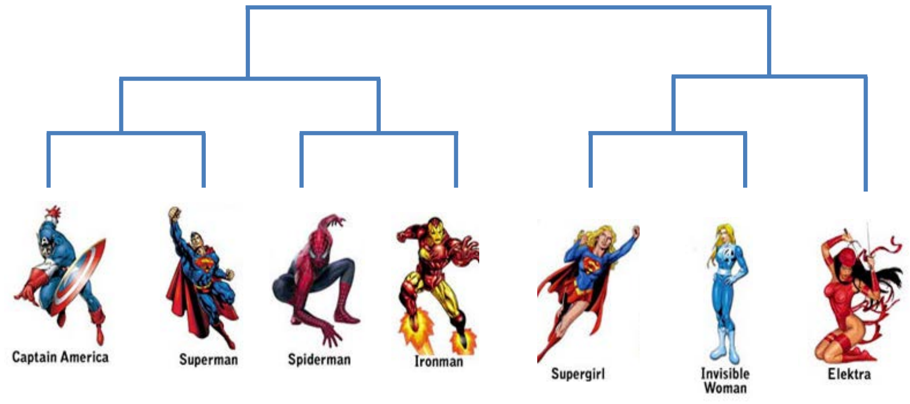
Clustering algorithms

- Typical partitional clustering algorithms
  - *k*-means clustering
    - Partition data by its closest mean
  - Gaussian Mixture Model
    - Consider variance within the cluster as well



Clustering algorithms

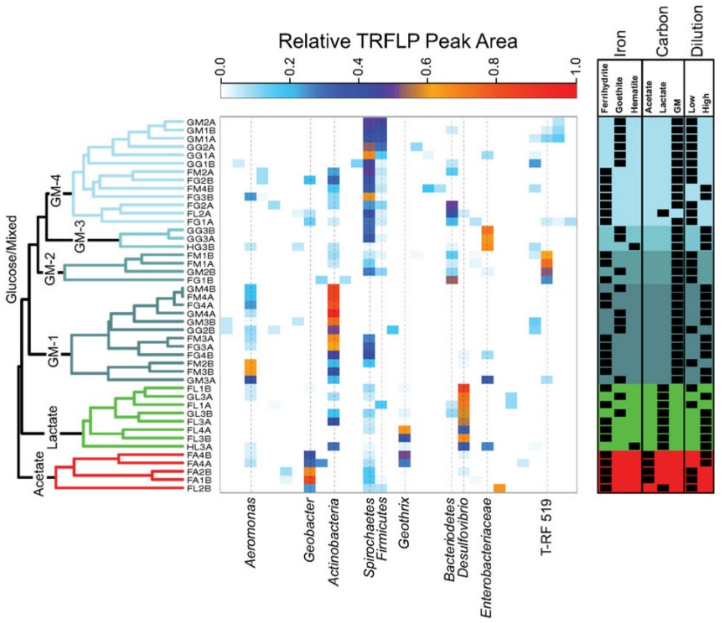
- Hierarchical clustering algorithms
  - Create a hierarchical decomposition of objects
  - Rich internal structure
    - No need to specify the number of clusters
    - Can be used to organize objects



Clustering algorithms

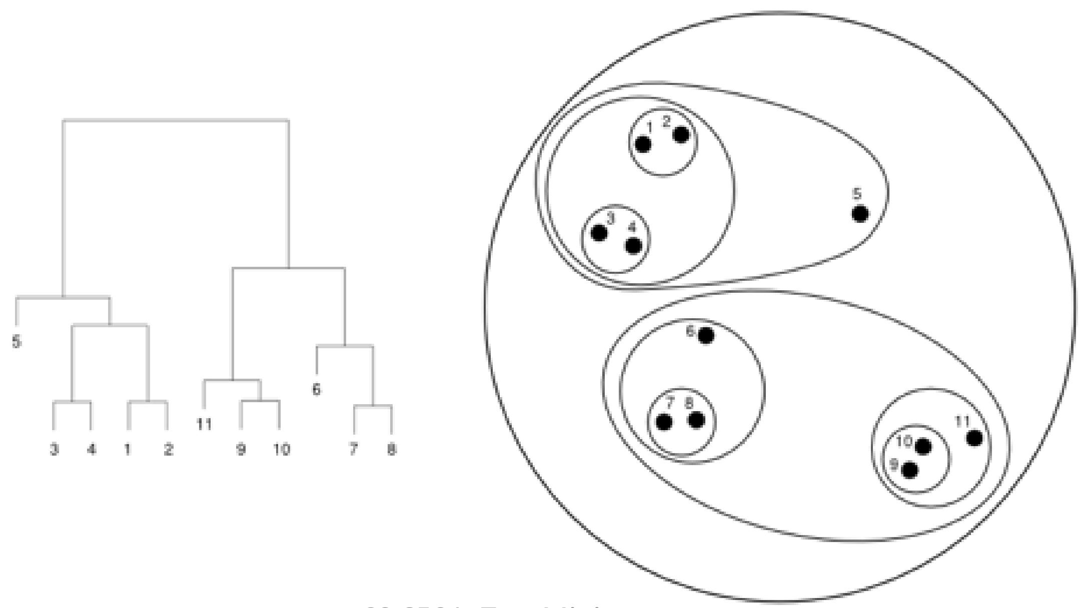
- Typical hierarchical clustering algorithms
  - Bottom-up agglomerative clustering
    - Start with individual objects as separated clusters
    - Repeatedly merge closest pair of clusters

Most typical usage:
gene sequence analysis



Clustering algorithms

- Typical hierarchical clustering algorithms
  - Top-down divisive clustering
    - Start with all data as one cluster
    - Repeatedly splitting the remaining clusters into two



Desirable properties of clustering algorithms

- Scalability
  - Both in time and space
- Ability to deal with various types of data
  - No/less assumption about input data
  - Minimal requirement about domain knowledge
- Interpretability and usability

Cluster validation

- Criteria to determine whether the clusters are meaningful
  – Internal validation
    - Stability and coherence
  – External validation
    - Match with known categories

Internal validation

- Coherence
  - Inter-cluster similarity v.s. intra-cluster similarity
  - Davies–Bouldin index

Evaluate every pair of clusters

- $DB = \frac{1}{k}\sum_{i=1}^{k} \max_{j \neq i}\left(\frac{\sigma_i + \sigma_j}{d(c_i, c_j)}\right)$
  - where k is total number of clusters, σ_i is average distance of all elements in cluster i, $d(c_i, c_j)$ is the distance between cluster centroid c_i and c_j.

We prefer smaller DB-index!

Internal validation

- Coherence
  - Inter-cluster similarity v.s. intra-cluster similarity
  - Dunn index
    - $D = \frac{\min_{1 \le i < j \le k} d(c_i, c_j)}{\max_{1 \le i \le k} \sigma_i}$ ***We prefer larger D-index!***
      - Worst situation analysis
- Limitation
  - No indication of actual application's performance
  - Bias towards a specific type of clustering algorithm if that algorithm is designed to optimize similar metric

External validation

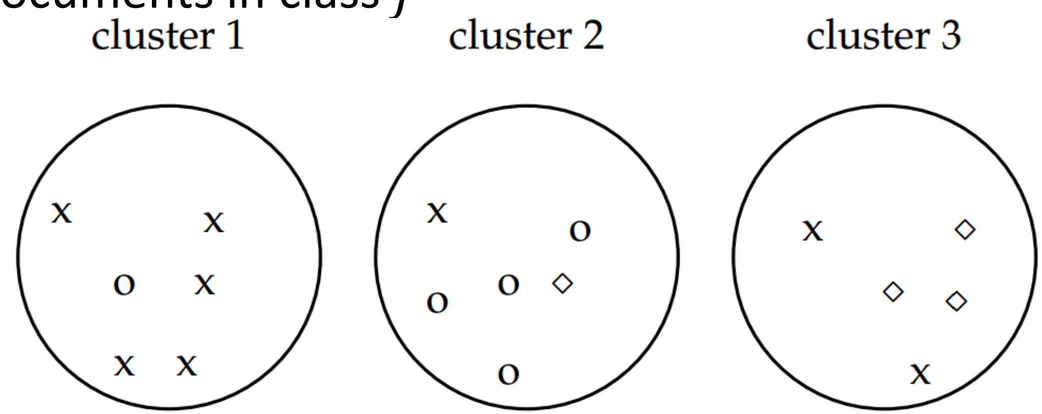
Required, might need extra cost

- Given class label Ω on each instance
  - Purity: correctly clustered documents in each cluster

Not a good metric if we assign each document into a single cluster

  - $purity(\Omega, C) = \frac{1}{N}\sum_{i=1}^{k} \max_{j} |c_i \cap w_j|$
    - where c_i is a set of documents in cluster i, and w_j is a set of documents in class j

$$purity(\Omega, C) = \frac{1}{17}(5 + 4 + 3)$$



External validation

- Given class label Ω on each instance
  - Normalized mutual information (NMI)
    - $NMI(\Omega, C) = \frac{I(\Omega,C)}{[H(\Omega)+H(C)]/2}$ ← *Normalization by entropy will penalize too many clusters*
      - where $I(\Omega, C) = \sum_i \sum_j P(w_i \cap c_j) \log \frac{P(w_i \cap c_j)}{P(w_i)P(c_j)}$, $H(\Omega) = -\sum_i P(w_i) \log P(w_i)$ and $H(\mathrm{C}) = -\sum_j P(c_j) \log P(c_j)$
    - Indicate the increase of knowledge about classes when we know the clustering results

External validation

- Given class label Ω on each instance
  - Rand index
    - Idea: we want to assign two documents to the same cluster if and only if they are from the same class
    - $RI = \frac{TP+TN}{TP+FP+FN+TN}$ ← Essentially it is like classification accuracy

	$w_i = w_j$	$w_i \neq w_j$
$c_i = c_j$	TP	FP
$c_i \neq c_j$	FN	TN

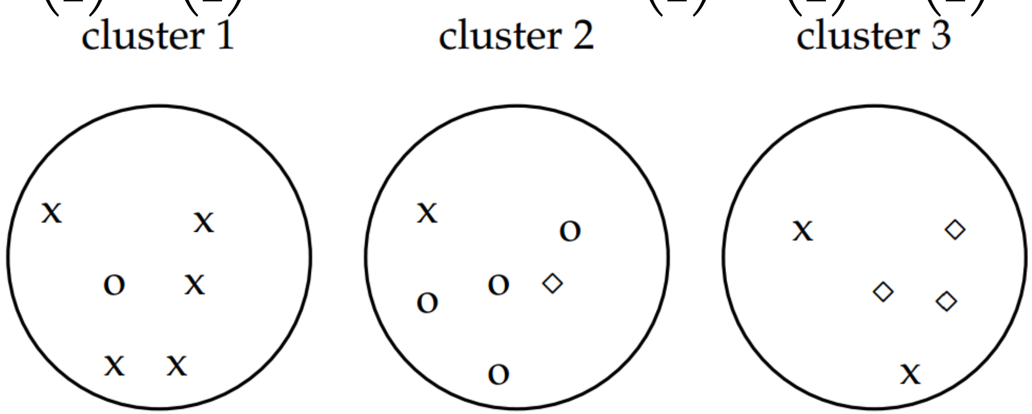
Over every pair of documents in the collection

External validation

- Given class label Ω on each instance
  - Rand index

	$w_i = w_j$	$w_i \neq w_j$
$c_i = c_j$	20	20
$c_i \neq c_j$	24	72

$$TP + FP = \binom{6}{2} + \binom{6}{2} + \binom{5}{2} = 40 \qquad TP = \binom{5}{2} + \binom{4}{2} + \binom{3}{2} + \binom{2}{2} = 20$$



External validation

- Given class label Ω on each instance
  - Precision/Recall/F-measure
    - Based on the contingency table, we can also define precision/recall/F-measure of clustering quality

	$w_i = \mathrm{w_j}$	$w_i \neq \mathrm{w_j}$
$c_i = c_j$	TP	FP
$c_i \neq c_j$	FN	TN

What you should know

- Unsupervised natural of clustering problem
  - Distance metric is essential to determine the clustering results
- Two basic categories of clustering algorithms
  - Partitional clustering
  - Hierarchical clustering
- Clustering evaluation
  - Internal v.s. external

Today's reading

- Introduction to Information Retrieval
  - Chapter 16: Flat clustering
    - 16.2 Problem statement
    - 16.3 Evaluation of clustering