Information Retrieval

Document Clustering

Yannis Tzitzikas

University of Crete

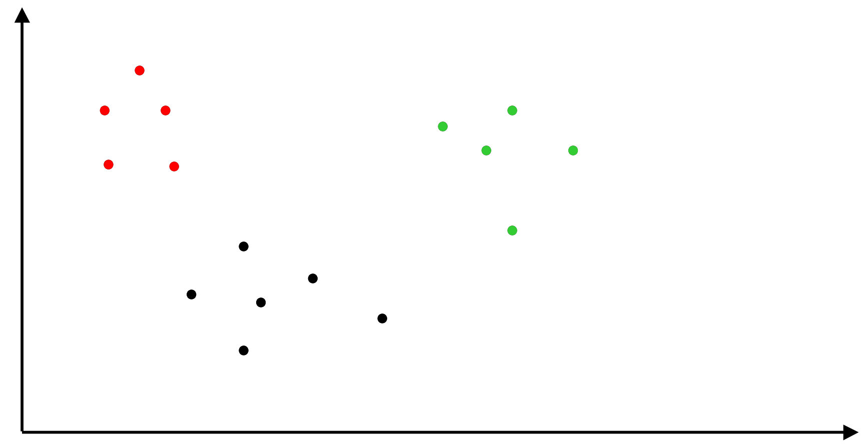
CS-463,Spring 05

Lecture : 7-8
Date : 15/17-3-2005

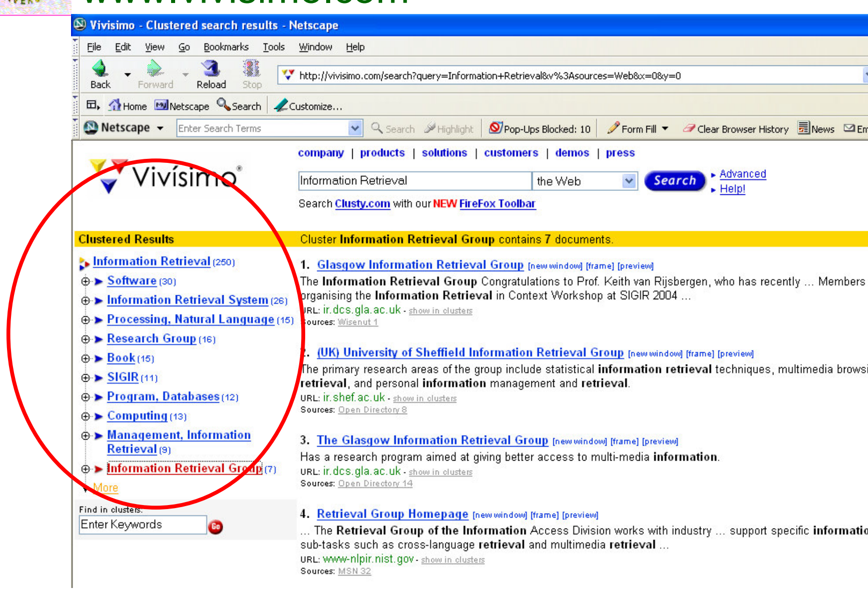
Clustering (Ομαδοποίηση): Εισαγωγή

- Στόχος: Ομαδοποίηση παρεμφερών αντικειμένων
  - Στην ΑΠ, για ομαδοποίηση εγγράφων και όρων
- Πολλές εφαρμογές
  - Ιατρικά και Κλινικά Δεδομένα, Φυτά, Ιστοσελίδες, Μεγάλα Εννοιολογικά Σχήματα κτλ
  - Πολλοί διαφορετικοί αλγόριθμοι και προσεγγίσεις
  - Graph theoretic, nearest means
- Τυπικά βασίζεται σε συγκρίσεις ζευγαριών χρησιμοποιώντας ένα μέτρο ομοιότητας
- Υπάρχουν πολλά δυνατά μέτρα ομοιότητας

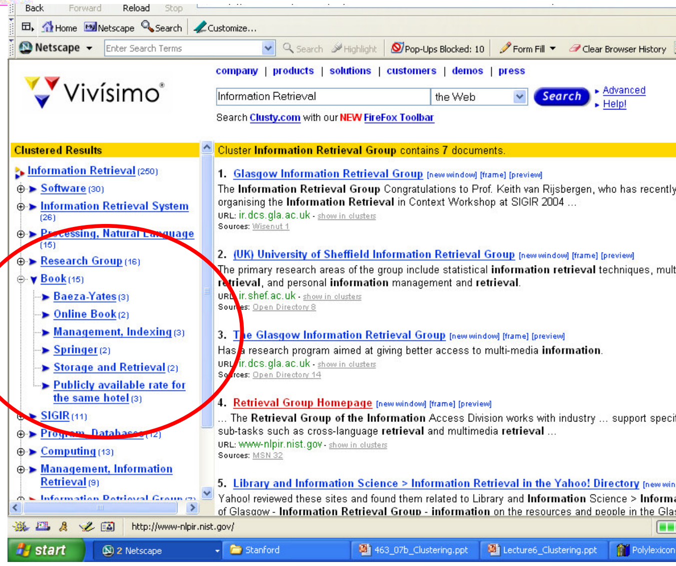
Clustering Example



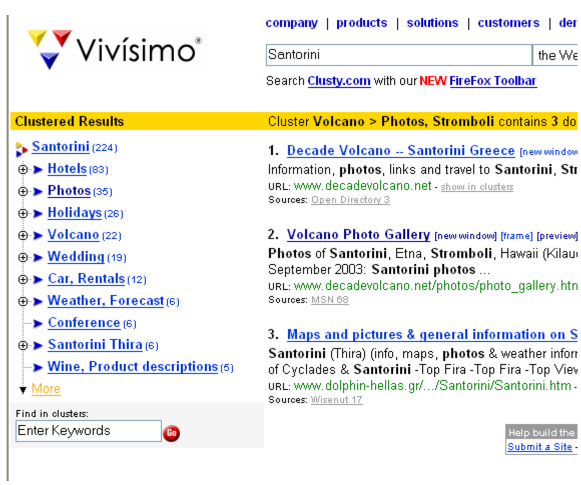
Πχ ομαδοποίησης αποτελεσμάτων www.vivisimo.com



Πχ ομαδοποίησης αποτελεσμάτων



q=Santorini



Τύποι Αλγορίθμων Ομαδοποίησης

- Ανάλογα με τη σχέση μεταξύ Ιδιοτήτων και Κλάσεων
  - Monothetic
  - Polythetic
- Ανάλογα με τη σχέση μεταξύ Αντικειμένων και Κλάσεων
  - Αποκλειστικά (exclusive)
  - Overlapping
- Ανάλογα με τη σχέση μεταξύ Κλάσεων
  - Με διάταξη (ιεραρχική)
  - Χωρίς διάταξη (απλή διαμέριση)

Monothetic vs. Polythetic

- Monothetic
  - Μια κλάση ορίζεται βάσει ενός συνόλου ικανών και αναγκαίων ιδιοτήτων που πρέπει να ικανοποιούν τα μέλη της (Αριστοτελικός ορισμός)
- Polythetic
  - Μια κλάση ορίζεται βάσει ενός συνόλου ιδιοτήτων Φ =φ1,…,φn, τ.ω.
    - Κάθε μέλος της κλάσης πρέπει να έχει ένα μεγάλο αριθμό των ιδιοτήτων Φ
    - Κάθε φ του Φ χαρακτηρίζει πολλά αντικείμενα
    - Δεν είναι αναγκαίο να υπάρχει μια φ που να ικανοποιείται από όλα τα μέλη της κλάσης
- Στην ΑΠ, έχει δοθεί έμφαση σε αλγόριθμους για αυτόματη παραγωγή polythetic classifications.

Monothetic vs. Polythetic

	A	B	C	D	E	F	G	H
1	+	+	+					
2	+	+		+				
3	+		+	+				
4		+	+	+				
5					+	+	+	
6					+	+	+	
7					+	+		+
8					+	+		+

Figure 3.1. An illustration of the difference between monothetic and polythetic

- 8 individuals (1-8) and 8 properties (A-H).
- The possession of a property is indicated by a plus sign. The individuals 1-4 constitute a polythetic group each individual possessing three out of four of the properties A,B,C,D.
- The other 4 individuals can be split into two monothetic classes {5,6} and {7,8}.

Μέτρα Συσχέτισης (Association)

- Μέτρα: Similarity, Association, Distance, Dissimilarity
  - Pairwise measure
  - Similarity increases as the number or proportion of shared properties increase
  - Typically normalized between 0 and 1
  - S(X,X)=1, S(X,Y)=S(Y,X)
- Παραδείγματα
  - Οι περισσότερες είναι κανονικοποιημένες εκδόσεις του |X ∩Y| ? t?? es? te?ί??? ???µ???? (e?? ????µe í eí a??µ????? ?????)
  - **Dice's coefficient** 2 |X ∩Y|/ |X| +|Y|
  - **Jaccard's coefficient** |X ∩Y|/ |X ∪ Y|
  - **Cosine correlation**
- ?e? ?p???e?t? «?a??te??» µ?t??

Παραδείγματα Μέτρων για Έγγραφα

- Dice's coefficient   2 $|X \cap Y| / |X| + |Y|$
- Jaccard's coefficient $|X \cap Y| / |X \cup Y|$
- Cosine correlation

$$\text{DiceSim (dj, dm)} = \boxed{\frac{2\sum_{i=1}^{t}(w_{ij} \cdot w_{im})}{\sum_{i=1}^{t} w_{ij}^{2} + \sum_{i=1}^{t} w_{im}^{2}}}$$

$$\text{JaccardSim (dj, dm)} = \boxed{\frac{\sum_{i=1}^{t}(w_{ij} \cdot w_{im})}{\sum_{i=1}^{t} w_{ij}^{2} + \sum_{i=1}^{t} w_{im}^{2} - \sum_{i=1}^{t}(w_{ij} \cdot w_{im})}}$$

$$\text{CosSim(dj, dm)} = \boxed{\frac{\vec{d}_j \cdot \vec{d}_m}{\left|\vec{d}_j\right| \cdot \left|\vec{d}_m\right|} = \frac{\sum_{i=1}^{t}(w_{ij} \cdot w_{im})}{\sqrt{\sum_{i=1}^{t} w_{ij}^{2} \cdot \sum_{i=1}^{t} w_{im}^{2}}}}$$

Clustering as Representation

- Clustering is unsupervised learning
  - Για εκμάθηση της υποκείμενης δομής και κλάσεων
- Clustering can be used to transform representations
  - Documents are represented by class membership as well as individual terms
- Can be viewed as dimensionality reduction
  - Ειδικά το term clustering
  - Latent Semantic Indexing, Factor Analysis είναι παρόμοιες τεχνικές

Clustering for <u>Efficiency</u>

- Η ιδέα:
  - 1/ Cluster documents,
  - 2/ Represent clusters by mean or average document,
  - 3/ **<u>compare query to cluster representatives</u>**
- Σχόλια:
  - Faster than sequential search
  - Not as fast as optimized inverted file
  - An inverted list is also a form of cluster

Clustering for <u>Effectiveness</u>

- By transforming representation, clustering may also result in more effective retrieval
- Retrieval of clusters makes it possible to retrieve documents that may not have many terms in common with the query
  - E.g. LSI

Document Clustering Approaches

- Graph Theoretic
  - Defines clusters based on a graph where documents are nodes and edges exist if similarity greater than some threshold
  - Require at least O(n2) computation
  - Naturally hierarchic (agglomerative)
  - Good formal properties
  - Reflect structure of data
- Based on relationships to cluster representatives or means
  - Define criteria for separability of cluster representatives
  - Typically have some measure of goodness of cluster
  - Require only O(n logn) or even O(n) computations
  - Tend to impose structure (e.g. number of clusters)
  - Can have undesirable properties (e.g. order dependence)
  - Usually produce partitions (no overlapping clusters)

Criteria of Adequacy for Clustering Methods

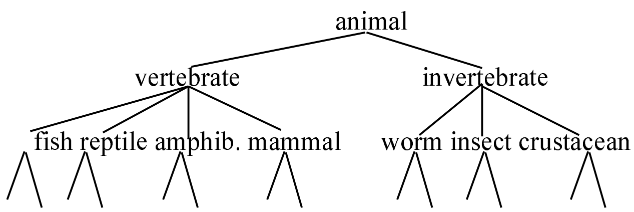
- The method produces a clustering which is unlikely to be altered drastically when further objects are incorporated (stable under growth)
- The method is stable in the sense that small errors in the description of objects lead to small changes in the clustering
- The method is independent of the initial ordering of the objects

Graph Theoretic Approaches

- Given a graph of objects connected by links that represent similarities greater than some threshold, the following cluster definitions are straightforward:
  - **Connected Component**: subgraph such that each node is connected to at least one other node in the subgraph and the set of nodes is maximal with respect to that property
    - Called **single link** clusters
  - **Maximal complete subgraph**: subgraph such that each node is connected to every other node in the subgraph (clique))
    - **Complete link** clusters
- Others are possible and very common:
  - **Average link**: each cluster member has a greater average similarity to the remaining members of the cluster than it does to all members of any other cluster

Hierarchical Clustering

- Build a tree-based hierarchical taxonomy (*dendrogram*) from a set of unlabeled examples.
- Recursive application of a standard clustering algorithm can produce a hierarchical clustering.



Hierarchical Clustering Methods

- Aglommerative (συσσώρευσης) (*bottom-up*) methods start with each example in its own cluster and iteratively combine them to form larger and larger clusters.
- *Divisive (διαίρεσης)* (*partitional, top-down*) separate all examples immediately into clusters.

An hierarchical clustering algorithm

1/ Βαλε κάθε έγγραφο σε ένα διαφορετικό cluster

2. Υπολόγισε την ομοιότητα μεταξύ όλων των ζευγαριών cluster

3. Βρες το ζεύγος {Cu,Cv} με την υψηλότερη (inter-cluster) ομοιότητα
4. Συγχώνευσε τα clusters Cu, Cv
5. Επανέλαβε (από το βήμα 2) έως ότου να καταλήξουμε να έχουμε 1 μόνο cluster
6. Επέστρεψε την ιεραρχία των clusters (το ιστορικό των συγχωνεύσεων)

An hierarchical clustering algorithm

1/ Βαλε κάθε έγγραφο σε ένα διαφορετικό cluster

C:=∅; For i=1 to n C:=C ∪ [di]

2. Υπολόγισε την ομοιότητα μεταξύ όλων των ζευγαριών cluster

Compute **SIM**(c,c') for each c, c' ∈ C

3. Βρες το ζεύγος {Cu,Cv} με την υψηλότερη (inter-cluster) ομοιότητα
4. Συγχώνευσε τα clusters Cu, Cv
5. Επανέλαβε (από το βήμα 2) έως ότου να καταλήξουμε να έχουμε 1 μόνο cluster
6. Επέστρεψε την ιεραρχία των clusters (το ιστορικό των συγχωνεύσεων)

An hierarchical clustering algorithm

1/ Βαλε <u>κάθε</u> έγγραφο σε ένα <u>διαφορετικό</u> cluster

$C:=\emptyset$; For i=1 to n $C:=C \cup [d_i]$

2. Υπολόγισε την <u>ομοιότητα</u> μεταξύ όλων των <u>ζευγαριών cluster</u>

Compute **SIM**(c,c') for each c, c' $\in$ C

sim(d,d') = CosineSim(d,d') or DiceSim(d,d') or JaccardSim(d,d')

3. Βρες το ζεύγος {Cu,Cv} με την <u>υψηλότερη</u> (inter-cluster) ομοιότητα
4. <u>Συγχώνευσε</u> τα clusters Cu, Cv
5. Επανέλαβε (από το βήμα 2) έως ότου να καταλήξουμε να έχουμε <u>1 μόνο cluster</u>
6. Επέστρεψε την ιεραρχία των clusters (το ιστορικό των συγχωνεύσεων)

An hierarchical clustering algorithm

1/ Βαλε <u>κάθε</u> έγγραφο σε ένα <u>διαφορετικό</u> cluster

$C:=\emptyset$; For i=1 to n $C:=C \cup [d_i]$

2. Υπολόγισε την <u>ομοιότητα</u> μεταξύ όλων των <u>ζευγαριών cluster</u>

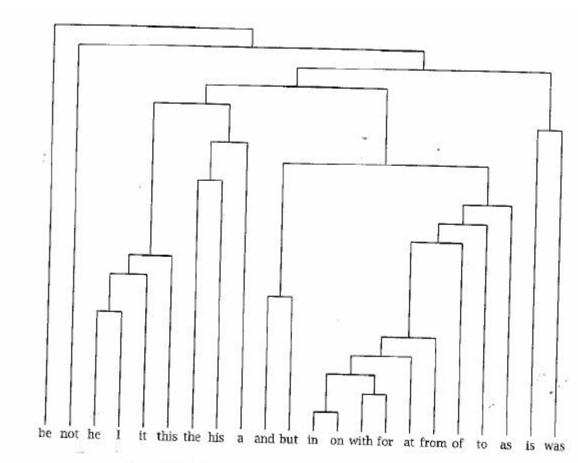
Compute **SIM**(c,c') for each c, c' $\in$ C

sim(d,d') = CosineSim(d,d') or DiceSim(d,d') or JaccardSim(d,d')

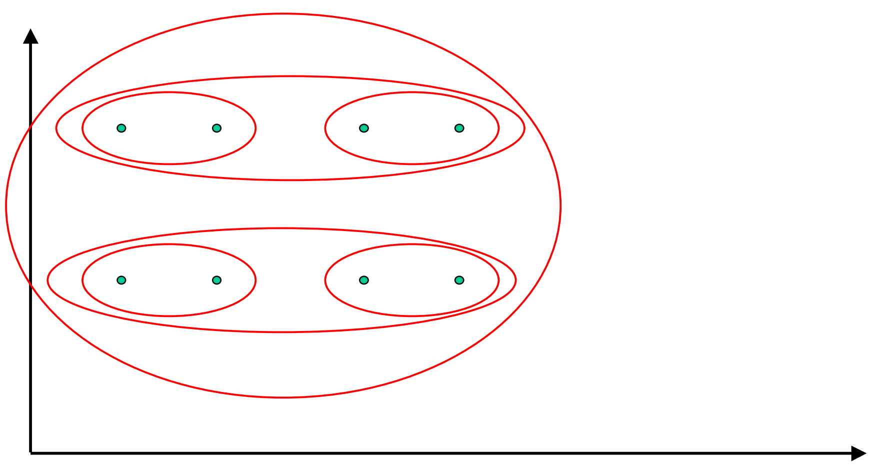
SIM(c,c')=
- *single link*: similarity of two <u>most similar</u>. = max{ sim(d,d') | $d \in c, d' \in c'$}
- *complete link*: similarity of two <u>least similar</u>. = min{ sim(d,d') | $d \in c, d' \in c'$}
- *average link*: <u>average</u> similarity b. = avg{ sim(d,d') | $d \in c, d' \in c'$}

3. Βρες το ζεύγος {Cu,Cv} με την <u>υψηλότερη</u> (inter-cluster) ομοιότητα
4. <u>Συγχώνευσε</u> τα clusters Cu, Cv
5. Επανέλαβε (από το βήμα 2) έως ότου να καταλήξουμε να έχουμε <u>1 μόνο cluster</u>
6. Επέστρεψε την ιεραρχία των clusters (το ιστορικό των συγχωνεύσεων)

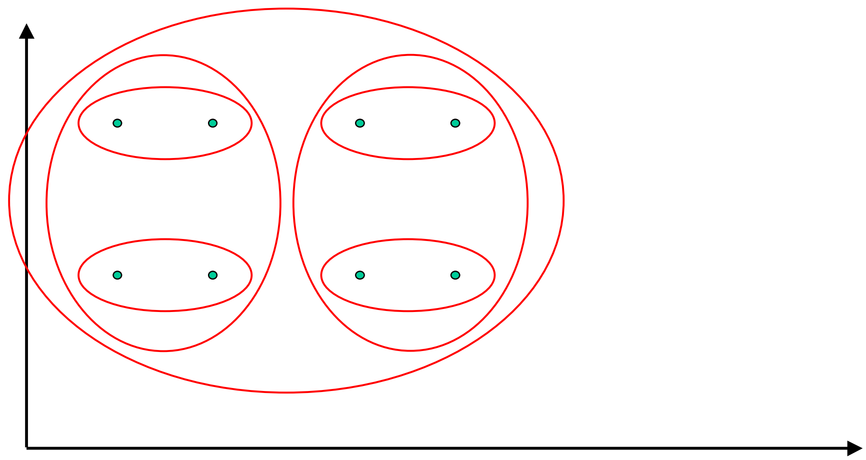
Dendogram or Cluster Hierarchy



Single Link Example



Complete Link Example



Σύγκριση

- Single-link
  - is provably the only method that satisfies criteria of adequacy
  - however it produces “long, straggly (ανάκατα) string” that are not good clusters
    - Only a single-link required to connect
- Complete link
  - produces good clusters (more “tight,” spherical clusters), but too few of them (many singletons)
- Average-link
  - For both searching and browsing applications, average-link clustering has been shown to produce the best overall effectiveness

Ward’s method
(an alternative to single/complete/average link)

- Cluster merging:
  - Merge the pair of clusters whose merger minimizes the increase in the total within-group error sum of squares, based on the Euclidean distance between centroids
- Remarks:
  - this method tends to create <u>symmetric hierarchies</u>

Computing the Document Similarity Matrix

Empty because sim(X,Y)=sim(Y,X)

$$
\begin{array}{cccccc}
d_1 & & & & & \\
d_2 & s_{21} & & & & \\
d_3 : & s_{31} & s_{32} & & & \\
: & : & : & & : & \\
d_n & s_{n1} & s_{n2} & \ldots & s_{n,n-1} & \\
 & d_1 & d_2 & \ldots. & d_{n-1} & d_n
\end{array}
$$

- Optimization: Compute sim(di,dj) only if di and dj have at least one term in common (otherwise it is 0)
  - This is done by exploiting the inverted index

Fast Partition Methods

Single Pass

- Assign the document d1 as the representative (**centroid,mean**) for c1
- For each di, calculate the similarity *Sim* with the representative for each existing cluster
- If SimMax is greater than threshold value *simThres*, add the document to the corresponding cluster and recalculate the cluster representative; otherwise use di to initiate a new cluster
- If a document di remains to be clustered, repeat

Fast Partition Methods

K-means (or reallocation methods)

- Select K cluster representatives
- For i = 1 to N, assign di to the most similar centroid
- For j = 1 to K, recalculate the cluster centroid cj
- Repeat the above steps until there is little or no change in cluster membership

- Issues:
  - How should K representatives be chosen?
  - Numerous variations on this basic method
    - cluster splitting and merging strategies
    - criteria for cluster coherence
    - seed selection

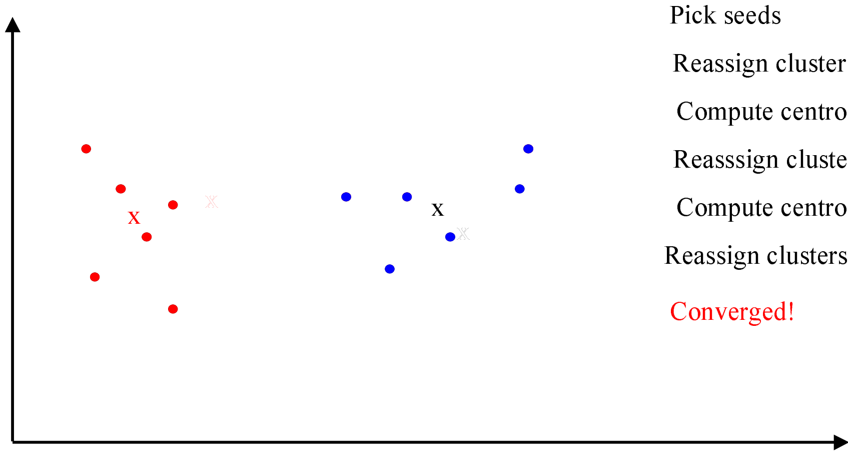
K-Means

- Assumes instances are real-valued vectors.
- Clusters based on *centroids*, *center of gravity*, or mean of points in a cluster, *c*:
  – For example, the centroid of (1,2,3), (4,5,6) and (7,2,6) is **(4,3,5).**

$$\vec{\mu}(\mathrm{c}) = \frac{1}{|c|}\sum_{\vec{x}\in c}\vec{x}$$

- Reassignment of instances to clusters is based on distance to the current cluster centroids.

K Means Example (K=2)



Pick seeds

Reassign clusters

Compute centroids

Reasssign clusters

Compute centroids

Reassign clusters

Converged!

Nearest Neighbor Clusters

- Cluster each document with its *k* nearest neighbors
- Produces overlapping clusters
- Called "star" clusters by Sparck Jones
- Can be used to produce hierarchic clusters
- cf. "documents like this" in web search

Complexity Remarks

- For computing the document matrix O(n^2)
- Simple reallocation clustering method with k clusters O(kn)
  - πιο γρήγορος από τους αλγορίθμους για ιεραρχική ομαδοποίηση
- Agglomerative or Divisive Hierarchical Clustering:
  - απαιτεί n-1 συγχωνεύσεις/διαιρέσεις
  - η πολυπλοκότητα του είναι τουλάχιστον O(n^2)

Cluster Searching
Document Retrieval from a Clustered Data Set

- *Top-down* searching:
  - start at top of cluster hierarchy,choose one of more of the best matching clusters to expand at the next level
    - tends to get lost
- *Bottom-up* searching:
  - create inverted file of “lowestlevel” clusters and rank them
    - more effective
    - indicates that highest similarity clusters (such as nearest neighbor) are the most useful for searching

Cluster Searching (II)

- After clusters are retrieved in order, documents in those clusters are ranked
- Cluster search produces similar level of effectiveness to document search, finds different relevant documents
- Cluster search can be modeled as a Bayesian classification problem with multiple categories
  - rank clusters by P(Ci|Q)

Human Clustering

- Ερωτήματα
  - Is there a clustering that people will agree on?
  - Is clustering something that people do consistently?
  - Yahoo suggests there's value in creating categories
    - Fixed hierarchy that people like
- "Human performance on clustering Web pages"
  - Macskassy, Banerjee, Davison, and Hirsh (Rutgers)
  - KDD 1998, and extended technical report
- Αποτελέσματα: Μάλλον δεν υπάρχει μεγάλη συμφωνία
    - γενικά προτίμηση σε μικρά clusters
    - άλλοι χρήστες προτιμούν/δημιουργούν επικαλυπτόμενα, άλλοι αποκλειστικά clusters
    - τα περιεχόμενα των clusters διέφεραν αρκετά
    - γενική ομαδοποίηση (ανεξαρτήτου επερώτησης) δεν φαίνεται να είναι πολύ χρήσιμη

Text Clustering

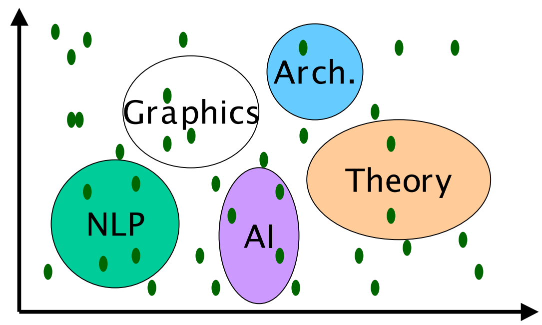
- HAC and K-Means have been applied to text in a straightforward way.
- Typically use ***normalized***, TF/IDF-weighted vectors and cosine similarity.
- Optimize computations for sparse vectors.
- Applications:
  - During retrieval, **add other documents** in the same cluster as the initial retrieved documents to improve recall.
  - **Clustering of results** of retrieval to present more organized results to the user (e.g. vivisimo search engine)
  - **Automated production of hierarchical taxonomies** of documents for browsing purposes (à la Yahoo & DMOZ).

RELATED ISSUES

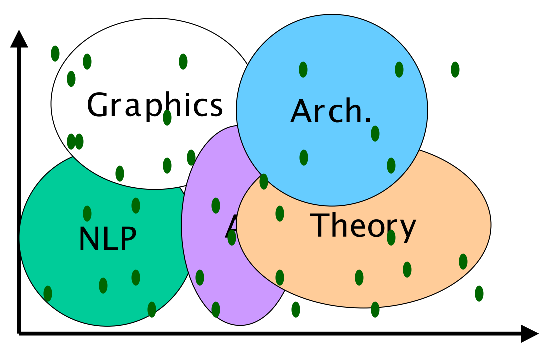
Clustering vs Classification

- Clustering
  - Unsupervised
  - Input
    - Clustering algorithm
    - Similarity measure
    - Number of clusters
  - No specific information for each document
- Classification
  - Supervised
  - Each document is labeled with a class
  - Build a classifier that assigns documents to one of the classes
- Two types of partitioning: supervised vs unsupervised

Text Classification Example



Text Classification Example



Supervised vs Unsupervised Learning

- This setup is called *supervised learning* in the terminology of Machine Learning
- In the domain of text, various names
  - **Text classification, text categorization**
  - **Document classification/categorization**
  - **“Automatic” categorization**
  - **Routing, filtering …**
- In contrast, the earlier setting of clustering is called *unsupervised learning*
  - Presumes no availability of training samples
  - Clusters output may not be thematically unified.

Text Categorization Examples

Assign labels to each document or web-page:

- Labels are most often **topics** such as Yahoo-categories

  e.g., "finance," "sports," "news>world>asia>business"
- Labels may be **genres**

  e.g., "editorials" "movie-reviews" "news“
- Labels may be **opinion**

  e.g., “like”, “hate”, “neutral”
- Labels may be **domain-specific binary**

  e.g., "interesting-to-me" : "not-interesting-to-me”

  e.g., “spam” : “not-spam”

  e.g., “contains adult language” :“doesn’t”

Classification Methods

- Manual classification
  - Used by Yahoo!, Looksmart, about.com, ODP, Medline
  - very accurate when job is done by experts
  - consistent when the problem size and team is small
  - difficult and expensive to scale
- Automatic document classification
  - Hand-coded **rule-based systems**
    - Used by spam filters, Reuters, CIA, Verity, …
      - E.g., assign category if document contains a given boolean combination of words
    - Commercial systems have complex query languages (everything in IR query languages + *accumulators*
    - Accuracy is often very high if a query has been carefully refined over time by a subject expert
    - Building and maintaining these queries is expensive

Classification Methods (II)

- Supervised learning of document-label assignment function
  - Many new systems rely on machine learning (Autonomy, Kana, MSN, Verity, Enkata, …)
    - k-Nearest Neighbors (simple, powerful)
    - Naive Bayes (simple, common method)
    - Support-vector machines (new, more powerful)
    - … plus many other methods
    - No free lunch: requires hand-classified training data
    - But can be built (and refined) by amateurs