Introduction to Information Retrieval
http://informationretrieval.org

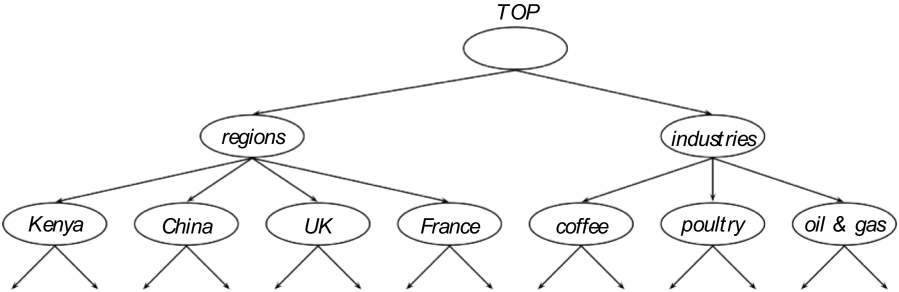
Hierarchical Clustering

(Chapter 17 of the textbook)

Slides borrowed from Hinrich Schütze with modifications

Hierarchical clustering

Our goal in hierarchical clustering is to create a hierarchy of clusters:



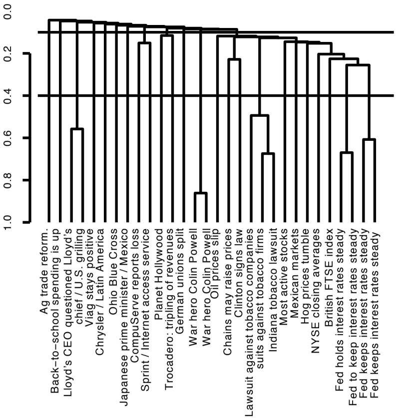
want to create this hierarchy automatically.
We can do this either top-down or bottom-up.

The best known bottom-up method is hierarchical agglomerative clustering (HAC).

HAC: Basic algorithm

- Start with each document in a separate cluster
- Then repeatedly merge the two clusters that are most similar
- Until there is only one cluster.
- The history of merging is a hierarchy in the form of a binary tree.
- The standard way of depicting this history is a dendrogram.

A dendrogram



- The history of mergers can be read off from bottom to top.
- The horizontal line of each merger tells us what the similarity of the merger was.
- We can cut the dendrogram at a particular point (e.g., at 0.1 or 0.4) to get a flat clustering.

Divisive clustering

- Divisive clustering is top-down.
- Alternative to HAC (which is bottom up).
- Divisive clustering:
  - Start with all docs in one big cluster
  - Then recursively split clusters
  - Eventually each node forms a cluster on its own.
- → Bisecting *K*-means (skip)
- For now: HAC (= bottom-up)

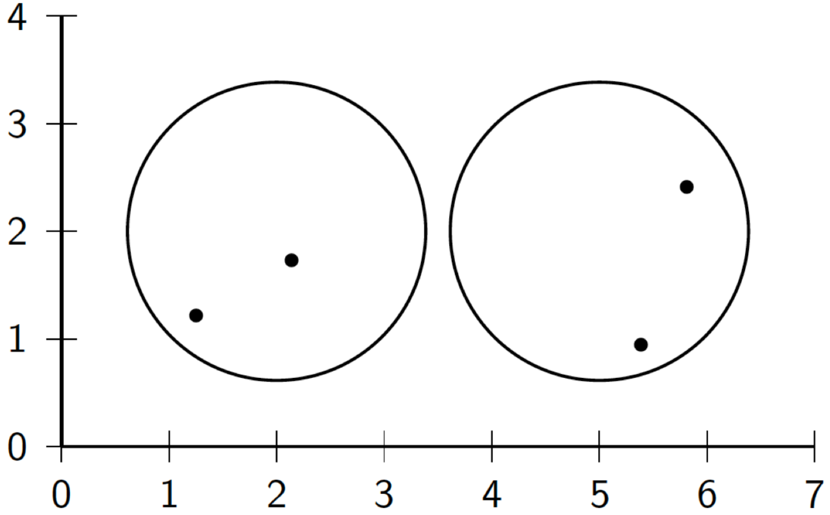
Hierarchical agglomerative clustering (HAC)

- HAC creates a hierarchy in the form of a binary tree.
- Assumes a similarity measure for determining the similarity of two clusters.
- Up to now, our similarity measures were for documents.
- We will look at four different cluster similarity measures.

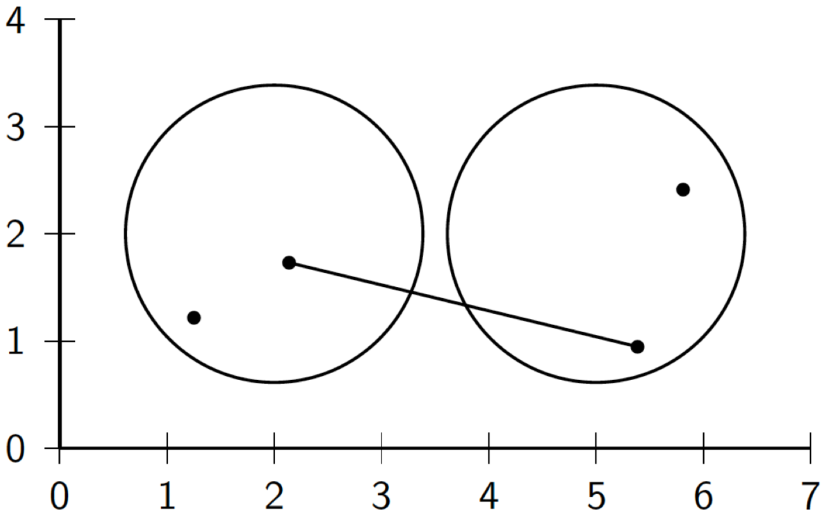
Key question: How to define cluster similarity

- Single-link: Maximum similarity
  - Maximum similarity of any two documents
- Complete-link: Minimum similarity
  - Minimum similarity of any two documents
- Centroid: Average "intersimilarity"
  - Average similarity of all document pairs (but excluding pairs of docs in the same cluster)
  - This is equivalent to the similarity of the centroids.
- Group-average: Average "intrasimilarity"
  - Average similary of all document pairs, including pairs of docs in the same cluster

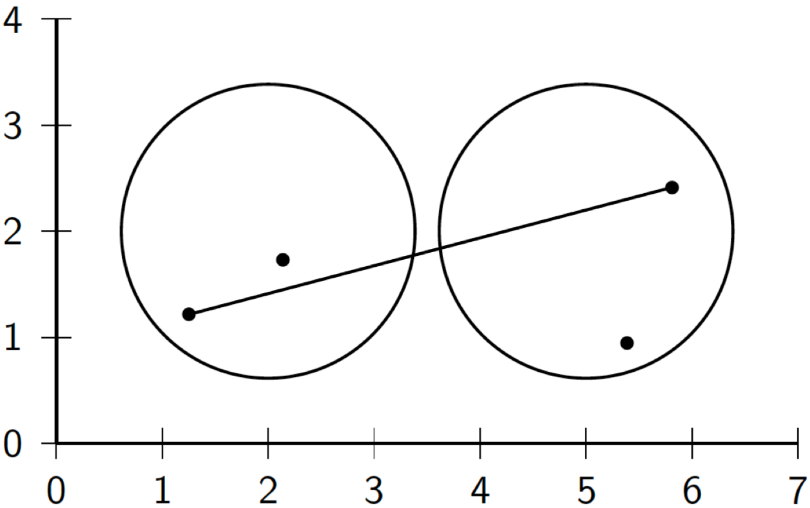
Cluster similarity: Example



Single-link: Maximum similarity

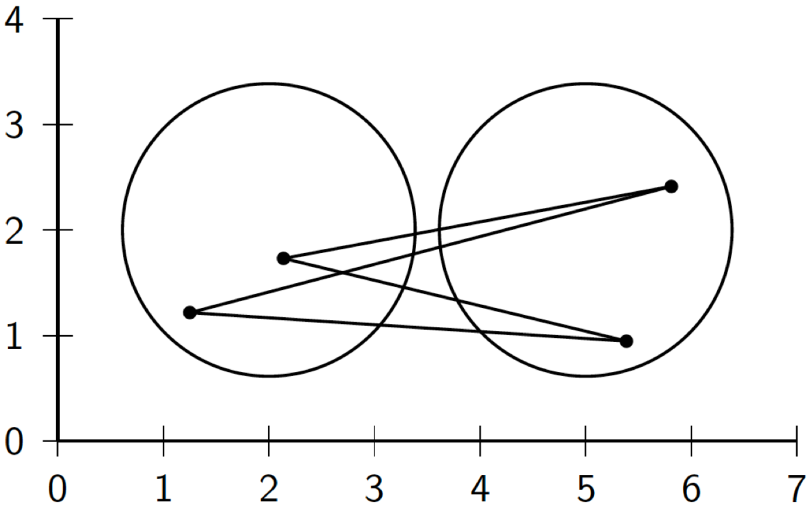


Complete-link: Minimum similarity



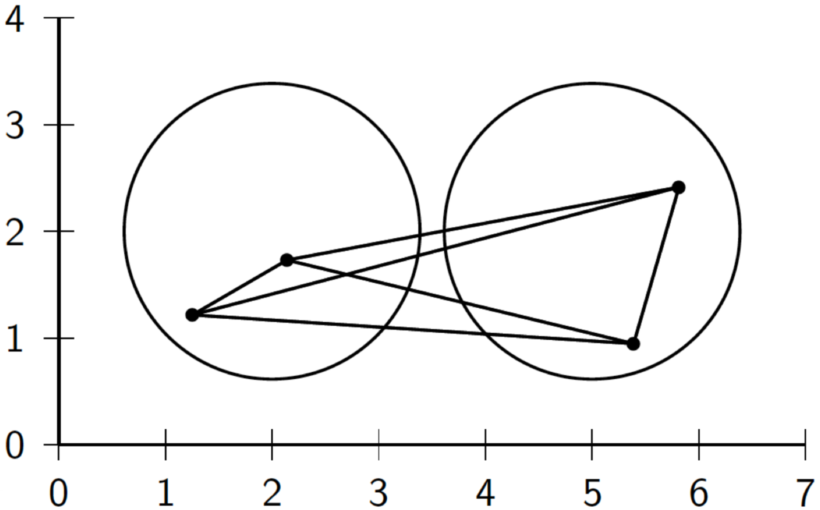
Centroid: Average intersimilarity

intersimilarity = similarity of two documents in different clusters



Group average: Average intrasimilarity

intrasimilarity = similarity of any pair, including cases where the two documents are in the same cluster



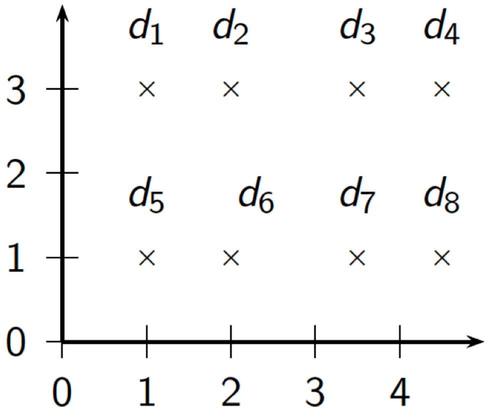
Single link HAC

- The similarity of two clusters is the maximum intersimilarity – the maximum similarity of a document from the first cluster and a document from the second cluster.

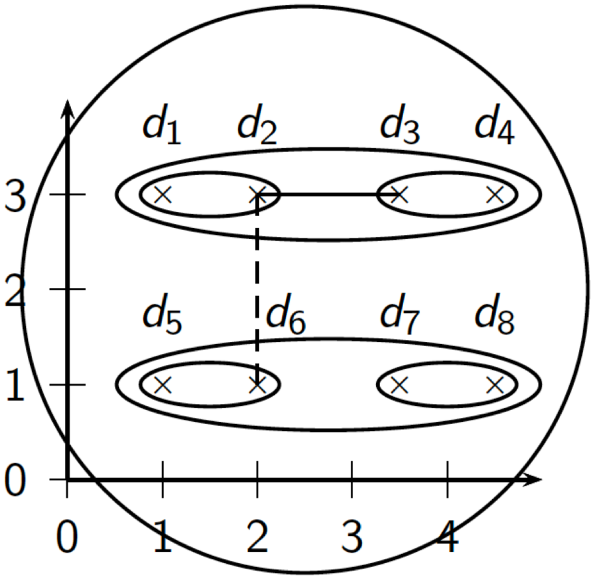
Complete link HAC

- The similarity of two clusters is the minimum intersimilarity – the minimum similarity of a document from the first cluster and a document from the second cluster.

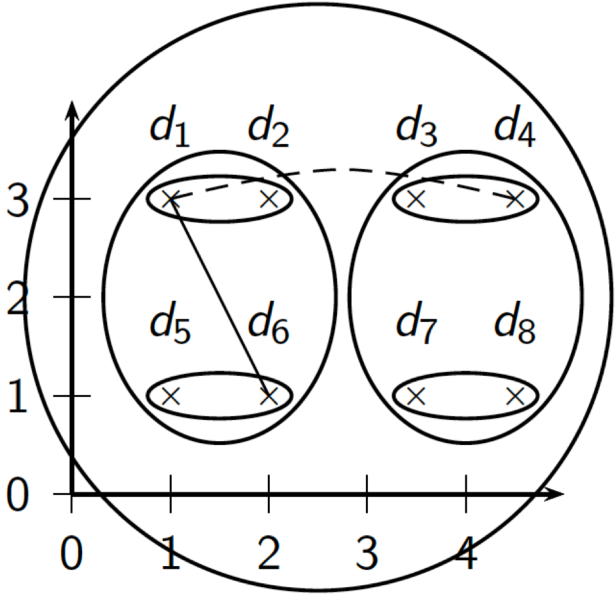
Exercise: Compute single and complete link clusterings



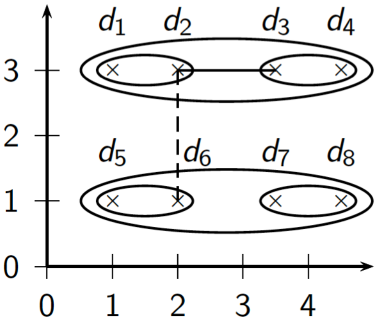
Single-link clustering

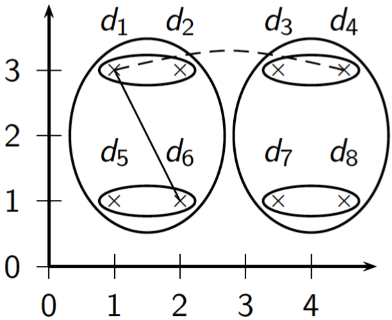


Complete link clustering

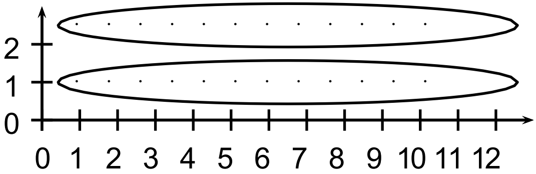


Single-link vs. Complete link clustering



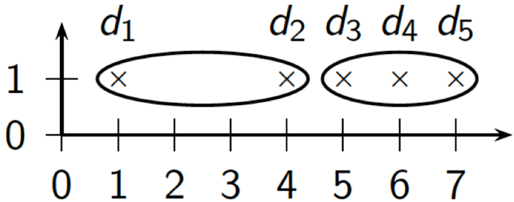


Single-link: Chaining



Single-link clustering often produces long, straggly clusters. For most applications, these are undesirable.

Complete-link: Sensitivity to outliers



- The complete-link clustering of this set splits d_2 from its right neighbors – clearly undesirable.
- The reason is the outlier d_1.
- This shows that a single outlier can negatively affect the outcome of complete-link clustering.
- Single-link clustering does better in this case.

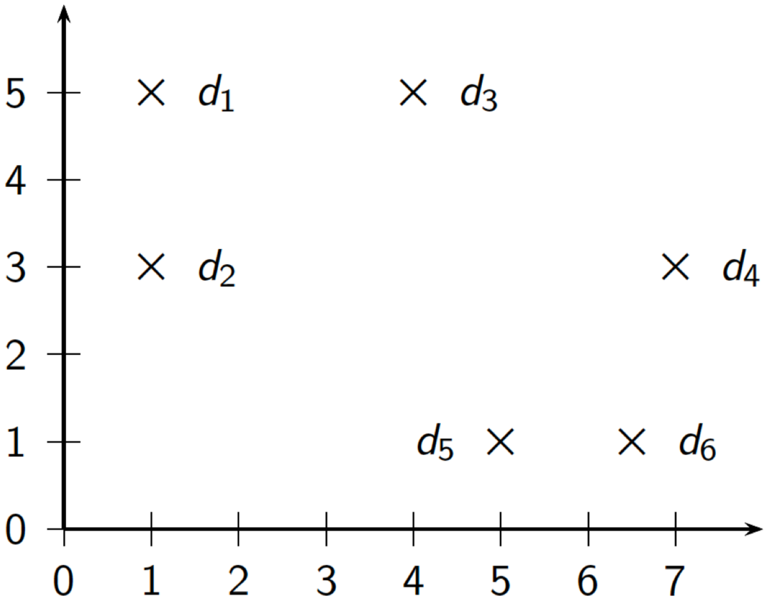
Centroid HAC

- The similarity of two clusters is the average intersimilarity – the average similarity of documents from the first cluster with documents from the second cluster.
- A naive implementation of this definition is inefficient ($O(N^2)$), but the definition is equivalent to computing the similarity of the centroids:

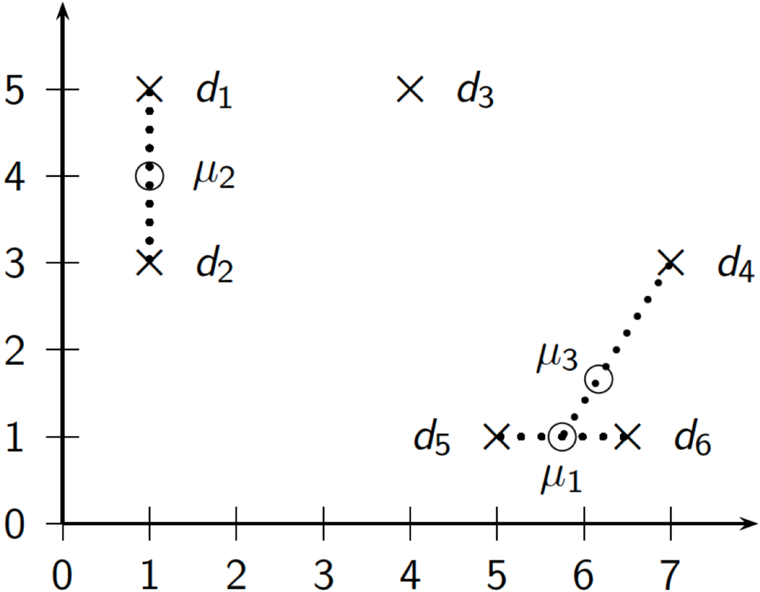
$$\text{SIM-CENT}(\omega_i, \omega_j) = \vec{\mu}(\omega_i) \cdot \vec{\mu}(\omega_j)$$

- Hence the name: centroid HAC
- Note: this is the dot product, not cosine similarity!

Exercise: Compute centroid clustering

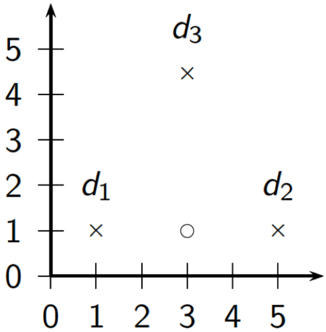


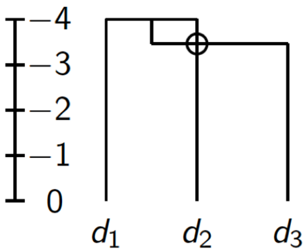
Centroid clustering



Inversion in centroid clustering

- In an inversion, the similarity increases during a merge sequence. Results in an "inverted" dendrogram.
- Below: Similarity of the first merger ($d_1 \cup d_2$) is -4.0, similarity of second merger (($d_1 \cup d_2$) $\cup$ d_3) is ≈ -3.5.





Inversions

- Hierarchical clustering algorithms that allow inversions are inferior.
- The rationale for hierarchical clustering is that at any given point, we've found the most coherent clustering for a given K.
- Intuitively: smaller clusters should be more coherent than larger clusters.
- An inversion contradicts this intuition: we have a large cluster that is more coherent than one of its subclusters.
- The fact that inversions can occur in centroid clustering is a reason not to use it.

Group-average agglomerative clustering (GAAC)

- GAAC also has an "average-similarity" criterion, but does not have inversions.
- The similarity of two clusters is the average intrasimilarity – the average similarity of all document pairs (including those from the same cluster).
- But we exclude self-similarities.

Group-average agglomerative clustering (GAAC)

- Again, a naive implementation is inefficient ($O(N^2)$) and there is an equivalent, more efficient, centroid-based definition:

$$\text{SIM-GA}(\omega_i, \omega_j) =$$

$$\frac{1}{(N_i + N_j)(N_i + N_j - 1)}[(\sum_{d_m \in \omega_i \cup \omega_j} \vec{d}_m)^2 - (N_i + N_j)]$$

- Again, this is the dot product, not cosine similarity.

Combination similarities of the four algorithms

clustering algorithm	$\mathrm{sim}(\ell, k_1, k_2)$
single-link	$\max(\mathrm{sim}(\ell, k_1), \mathrm{sim}(\ell, k_2))$
complete-link	$\min(\mathrm{sim}(\ell, k_1), \mathrm{sim}(\ell, k_2))$
centroid	$(\frac{1}{N_m}\vec{v}_m) \cdot (\frac{1}{N_\ell}\vec{v}_\ell)$
group-average	$\frac{1}{(N_m+N_\ell)(N_m+N_\ell-1)}[(\vec{v}_m + \vec{v}_\ell)^2 - (N_m + N_\ell)]$

Which HAC clustering should I use?

- Don't use centroid HAC because of inversions.
- In most cases: GAAC is best since it isn't subject to chaining or sensitivity to outliers.
- However, we can only use GAAC for vector representations.
- For other types of document representations (or if only pairwise similarities for documents are available): use complete-link.
- There are also some applications for single-link (e.g., duplicate detection in web search).

What to do with the hierarchy?

- Use as is (e.g., for browsing as in Yahoo hierarchy)
- Cut at a predetermined threshold
- Cut to get a predetermined number of clusters *K*
  - Ignores hierarchy below and above cutting line.

Flat or hierarchical clustering?

- For high efficiency, use flat clustering (or perhaps bisecting *k*-means)
- For deterministic results: HAC
- When a hierarchical structure is desired: hierarchical algorithm
- HAC also can be applied if *K* cannot be predetermined (can start without knowing *K*)

Major issue in clustering – labeling

- After a clustering algorithm finds a set of clusters: how can they be useful to the end user?
- We need a pithy label for each cluster.
- For example, in search result clustering for “jaguar”, The labels of the three clusters could be “animal”, “car”, and “operating system”.
- Topic of this section: How can we automatically find good labels for clusters?

Exercise

- Come up with an algorithm for labeling clusters
- Input: a set of documents, partitioned into K clusters (flat clustering)
- Output: A label for each cluster
- Part of the exercise: What types of labels should we consider? Words?

Cluster labeling: Example

		labeling method		
	# docs	centroid	mutual information	title
4	622	oil plant mexico production crude **power000refinerygas** bpd	plant oil production **barrels** crude bpd mexico **dolly capacitypetroleum**	MEXICO: Hurricane Dolly heads for Mexico coast
9	1017	police security **russian** people military peace killed told **groznycourt**	police killed military security peace told **troops forcesrebels** people	RUSSIA: Russia's Lebed meets rebel chief in Chechnya
10	1259	00 000 tonnes traders futures wheat prices **centsseptember** tonne	**delivery** traders futures tonne tonnes **desk** wheat prices 000 00	USA: Export Business - Grain/oilseeds complex

- Three methods: most prominent terms in centroid, differential labeling using MI, title of doc closest to centroid
- All three methods do a pretty good job.