Search Engines

WS 2009 / 2010

Lecture 13, Thursday February 4th, 2010

(Hierarchical Clustering)

Prof. Dr. Hannah Bast
Chair of Algorithms and Data Structures
Department of Computer Science
University of Freiburg

UNI FREIBURG

Overview of Today's Lecture

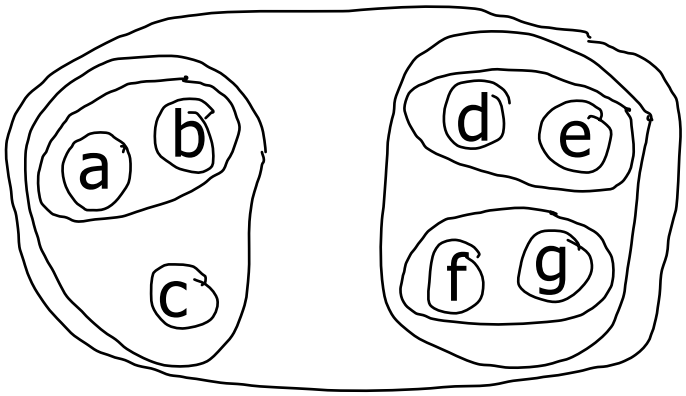
UNI FREIBURG

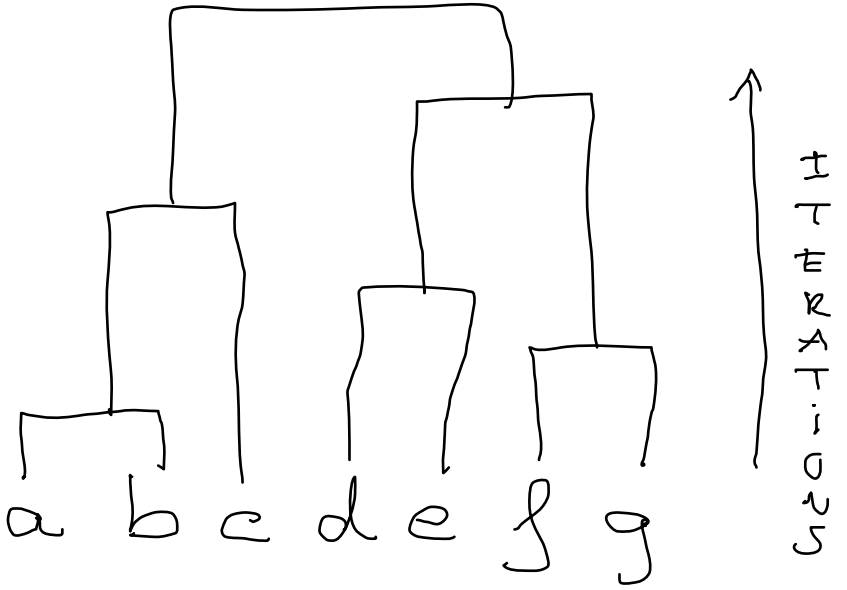
- Learn about Hierarchical Clustering
  - what it is
  - how it compares to "flat" clustering (like k-means)

  - was planned for last lecture
  - but due to the usual technical problems we dropped it
  - enough material for a whole own lecture though

Hierarchical Clustering

- General bottom-up idea:
  - start with clustering, where each point is its own cluster
  - iteratively merge the two clusters that are "most similar"
  - natural visualization of hierarchy as a dendrogram

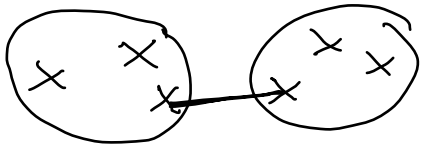




Which Clusters To Merge

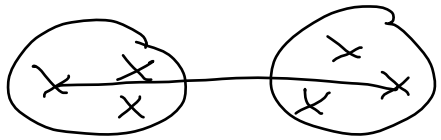
- Similarity measure between clusters $sim(C_i, C_j)$
  - in each step merge C_i and C_j with largest $sim(C_i, C_j)$
- Four common similarity measures
  - Single-Link: similarity of closest points

$$sim(C_i, C_j) = \max_{x \in C_i, y \in C_j} sim(x, y)$$



  - Complete-Link: similarity of farthest points

$$sim(C_i, C_j) = \min_{x \in C_i, y \in C_j} sim(x, y)$$

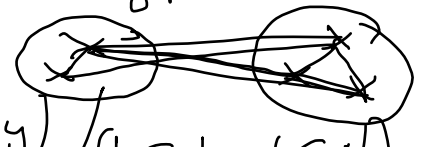


  - Centroid: average inter-similarity

$$sim(C_i, C_j) = \sum_{x \in C_i} \sum_{x \in C_j} sim(x, y) / |C_i| \cdot |C_j|$$

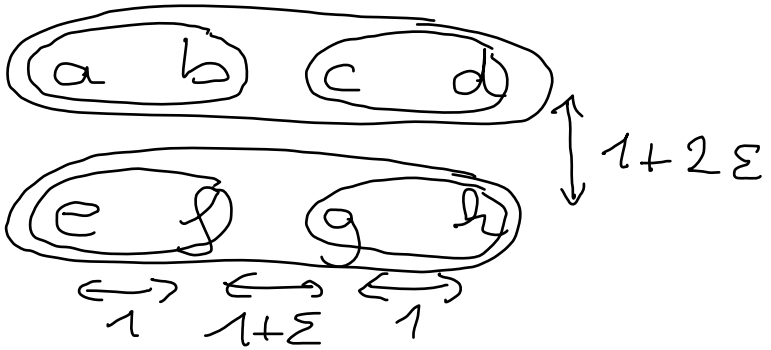
  - Group-Average: average of all similarities

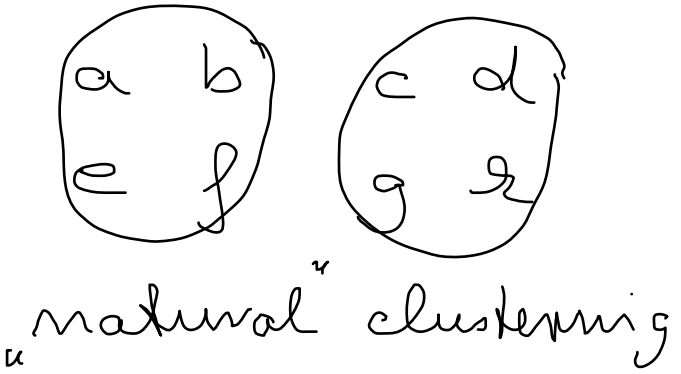
$$sim(C_i, C_j) = \sum_{x \in C_i \cup C_j} \sum_{\substack{y \in C_i \cup C_j \\ y \neq x}} sim(x, y) / (|C_i| + |C_j|) \cdot (|C_i| + |C_j| - 1)$$



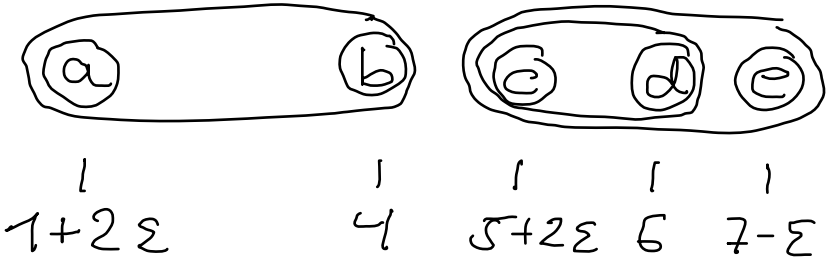
Single-Link and Complete-Link

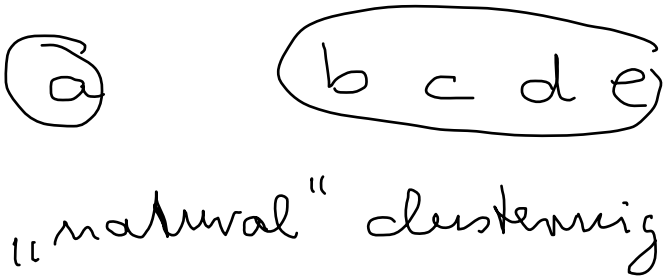
- Single-Link Problem
  - only the closest pair counts → tendency to straggly clusters





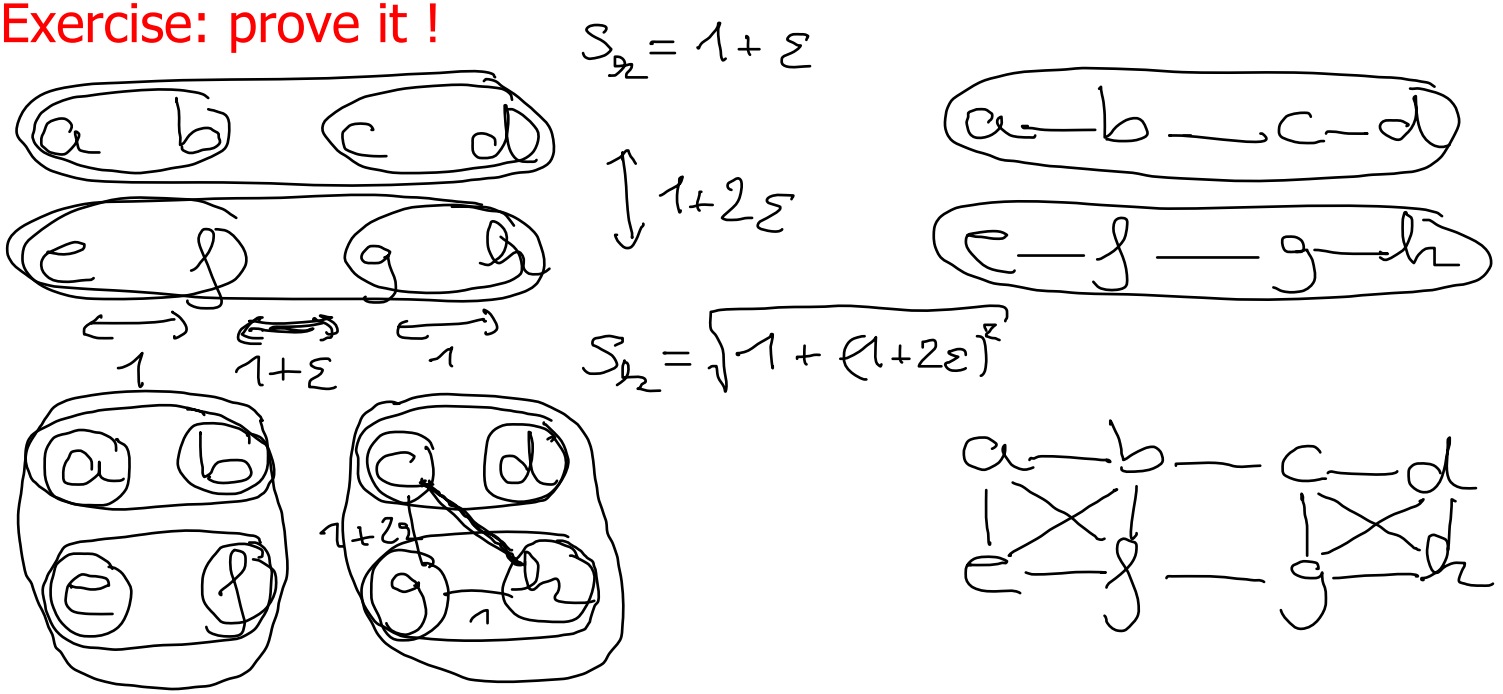
- Complete-Link Problem
  - high sensitivity to outliers, even to single one





Why the names Single / Complete Link?

- Graph-theoretic interpretation
  - let $s_k = sim(C_i, C_j)$ in k-th merging step
  - let G_k be the graph with an edge between all points with $s \geq s_k$
  - then single-link clusters = connected components of G_k
  - and complete-link clusters = maximal cliques of G_k
  - Exercise: prove it !



Hierarchical Clustering — Algorithm

UNI FREIBURG

- Again, code live in a VNC session
  - again with points = numbers
  - pay attention, you will need this for the exercises

Hierarchical Clustering — Time Complexity

UNI FREIBURG

- Naive algorithm
  - for full hierarchy, time complexity is $O(n^3)$
  - if we proceed for k iterations, still $O(k \cdot n^2)$
  - n^2 is prohibitive for large data (think of n = 1 million)
- Improvement
  - using a priority queue we can achieve $O(k \cdot n \cdot \log n)$
    - Exercise: implement for complete-link
  - this is ok; recall that k-means needs $O(I \cdot k \cdot n)$

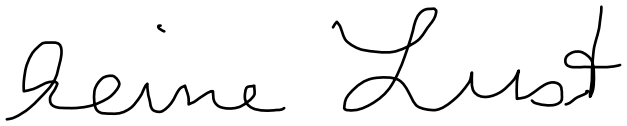
Hierarchical Clustering — Time Compl. 2

- Further improvement
  - for single-link we can even achieve $O(k \cdot n)$
    - that is, each iteration in linear time
  - because single-link is best-merge persistent
    - let C_i be the most similar cluster for C_k
    - assume C_i gets merged with $C_j \neq C_k$
    - after that $C_i \cup C_j$ is the most similar cluster for C_k
  - Exercise: implement single link using NBM-array

    NBM = next best merge
  - Exercise: show that complete-link is not best-merge persistent

Hierarchical Clustering — Time Compl. 3

- What is the cost of the similarity computations?
  - for single-link and complete-link we have to compute the n^2 similarities of all point pairs only once at the beginning
  - for group-average hierarchical clustering, efficient for cosine similarity (we need: distributivity of + and ·)



References

- Already given in slides from last lecture
  - The Wikipedia articles is ok
    http://en.wikipedia.org/wiki/Hierarchical_clustering
  - Here is the textbook which I also consulted
    Introduction to Information Retrieval