Lecture 10: Divide

”What are natural communities, divisions? Who is likely to love/hate my blog?"

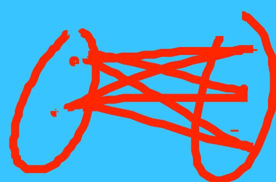
COMS 4995-1: Introduction to Social Networks

Tuesday, October 9th

Outline

* Groups, Clusters, Communities
* Extraction
  * Modularity, Conductance, Hierarchy
* Validation
  * Impact of size, Ground truth
* Evolution & Community model
* Study of network with positive & negative links

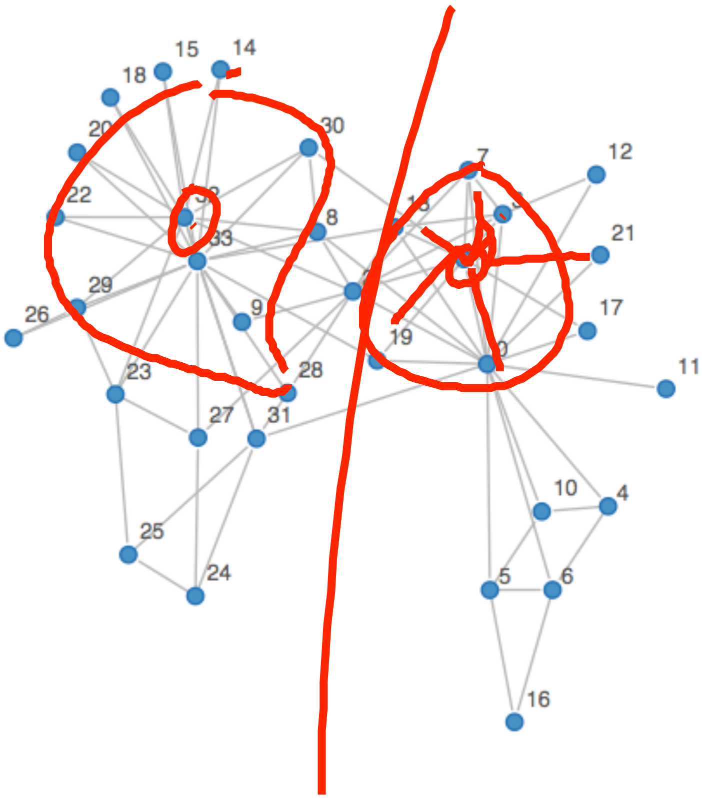
1999: Cyber-communities

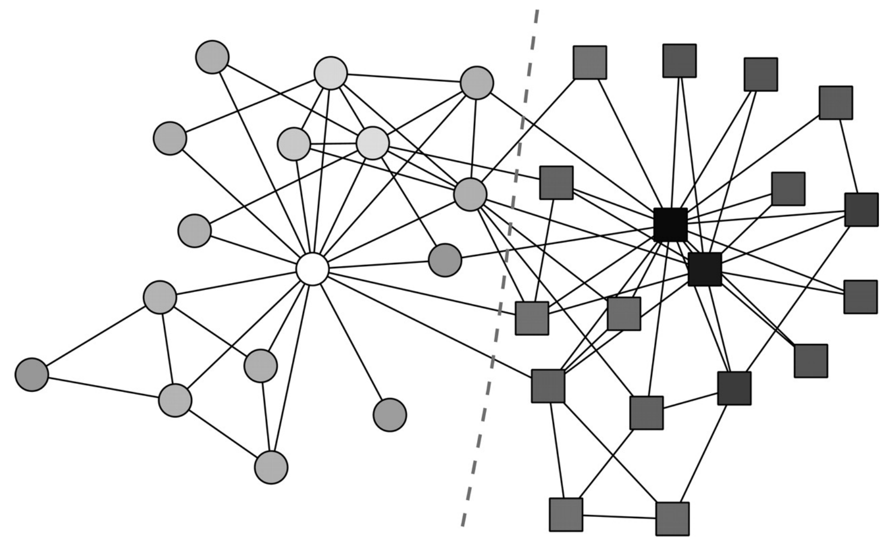
DATA SET: Webgraph from 1998 OBJECTIVE Find (i,j) cores (compl. bipartite graph) (2,3) 	EMPIRICAL OBSERVATIONS On 200 (3,3) cores and 200 (3,5) cores • 30% are fossilized (half life of webpage~ 6months) • 5% are not cogent • 56% not in yahoo
ALGORITHM Remove duplicate, Remove large in-deg, Iterative pruning (with smart impl.) Inclusion-Exclusion	ANALYSIS

Kumar, R., Raghavan, P., Rajagopalan, S., & Tomkins, A. (1999). Trawling the Web for emerging cyber-communities. *Computer Networks*

What is a community?

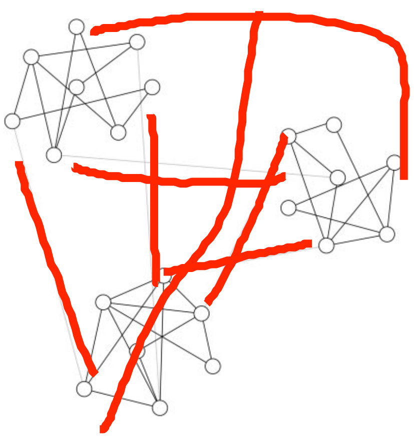
* Where is the community

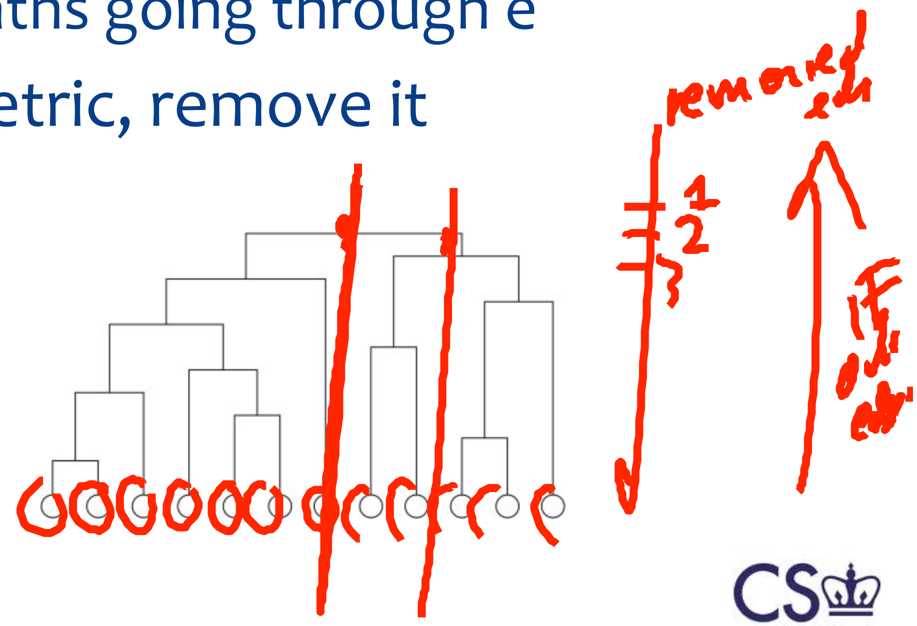




Possible metric (1) Division

* Dividing a network
* Use Betweenness and division
  - Defines the edge-betweeness of e=(u,v) as
    * The number of shortest paths going through e
  - Find the edge with max metric, remove it





Possible metric (2) Merge

* Modularity and agglomeration
  - fraction of edges within cluster
  - Minus the expected

$\sum_{ij} A_{ij} - \sum_{ij} \frac{k_i k_j}{2m}$

$k_i = \sum_{j \neq i} A_{ij}$

#edges

$$Q = \frac{1}{2m} \sum_{i,j} \left[A_{ij} - \frac{k_i k_j}{2m} \right] \delta(c_i, c_j) \quad \in [-1, 1]$$

* Merge clusters, how is Q varying when i moves to C

$$\Delta Q = \left[\frac{\sum_{\text{in}} + 2k_{i,\text{in}}}{2m} - \left(\frac{\sum_{\text{tot}} + k_i}{2m} \right)^2 \right] - \left[\frac{\sum_{\text{in}}}{2m} - \left(\frac{\sum_{\text{tot}}}{2m} \right)^2 - \left(\frac{k_i}{2m} \right)^2 \right]$$

CS@CU

Possible metric (3) Conductance

* Minimizing the conductance of a partition Bisection

$$\phi(S) = \frac{\sum_{i\in S, j\notin S} A_{ij}}{\min\{A(S), A(\bar{S})\}} \qquad A(S) = \sum_{i\in S}\sum_{j\in V} A_{ij} = \sum_{i\in S} d_i$$

* Efficient techniques to divide a graph in 2
  - Could be using relaxation method
  - even proved to approximate NP-hard problems
* Then apply the same algorithm recursively