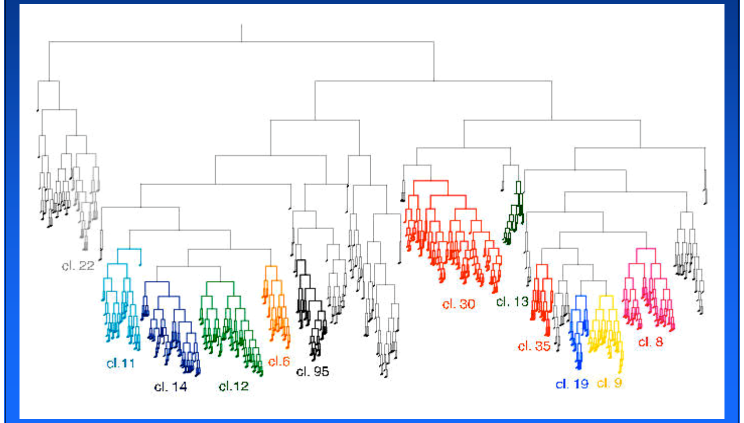
Semi-Supervised Learning, Clustering with User Feedback, and Meta Clustering

What is Clustering?

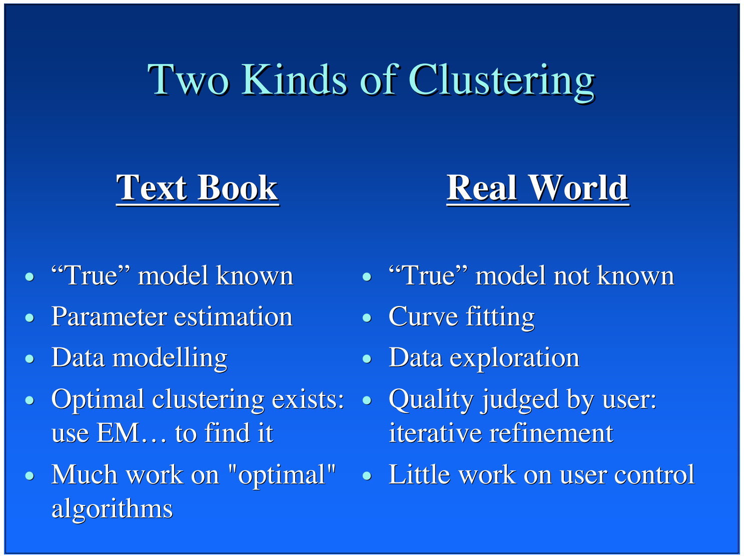
- Finding groups of similar objects in data
  - Clustering people with similar characteristics
    - Activities
    - Network of associations
    - Educational, socio-economic, background
    - Beliefs and behaviors
  - Clustering text/documents with similar characteristics
    - By content
    - By document type
    - By document intent
    - By intended audience
  - Clustering network events
    - By intent: attack vs. intrusion vs. denial of service vs. normal
    - By type: port scan vs. probe vs. …

What is Clustering?



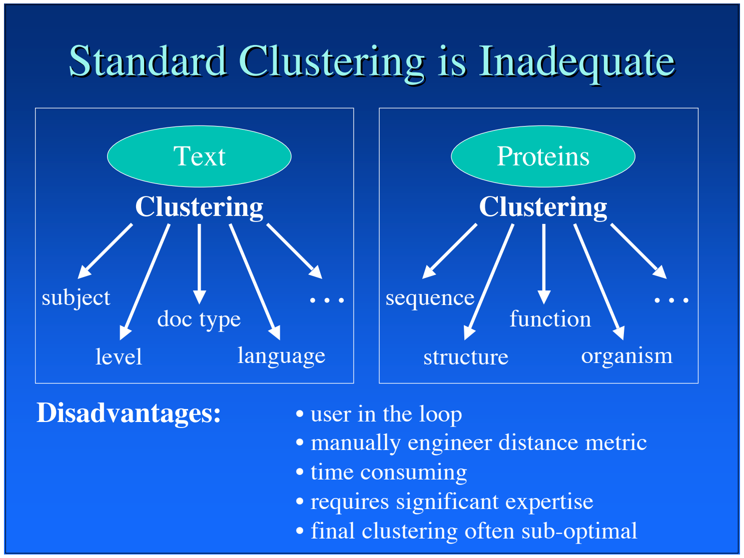
Why Clustering?

- Data exploration
  - Our capacity to collect data has outstripped our capacity to understand/interpret the data
  - Chicken and the egg problem with new data:
    - Don't know what you're looking for until you understand data
    - Can't understand data until you know what to look for
  - Easier to find patterns in groups of objects than in single objects
  - As data grows bigger, but human brain remains fixed, must present experts with less raw, more processed data
- Focused search and data analysis
  - soft/fuzzy/approximate/smart queries
- Efficient transmission, presentation, summarization



Two Kinds of Clustering

Text Book	**Real World**
• "True" model known	• "True" model not known
• Parameter estimation	• Curve fitting
• Data modelling	• Data exploration
• Optimal clustering exists: use EM… to find it	• Quality judged by user: iterative refinement
• Much work on "optimal" algorithms	• Little work on user control

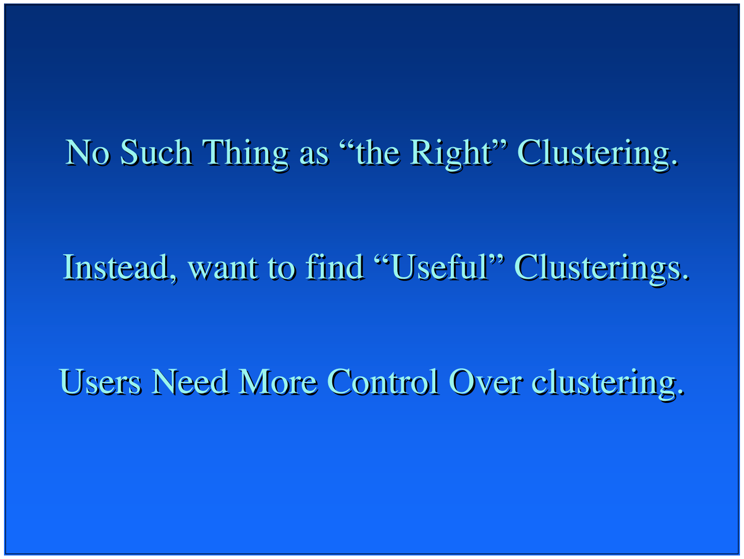


Standard Clustering is Inadequate

Text
Clustering
subject
level
doc type
language
. . .
Proteins
Clustering
sequence
structure
function
organism
. . .

Disadvantages:

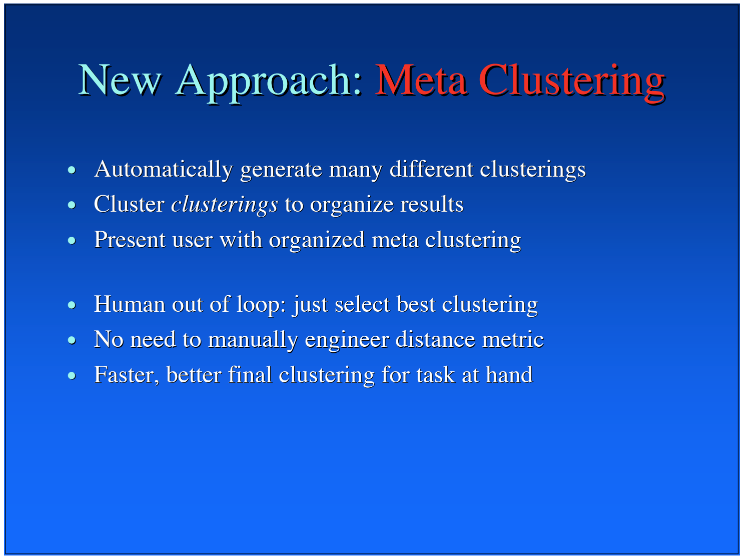
- user in the loop
- manually engineer distance metric
- time consuming
- requires significant expertise
- final clustering often sub-optimal



No Such Thing as "the Right" Clustering.

Instead, want to find "Useful" Clusterings.

Users Need More Control Over clustering.



New Approach: Meta Clustering

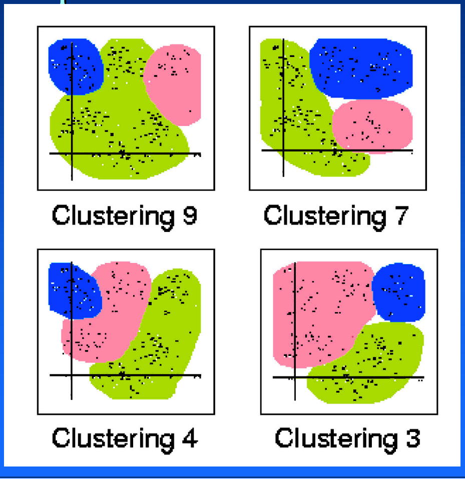
- Automatically generate many different clusterings
- Cluster *clusterings* to organize results
- Present user with organized meta clustering

- Human out of loop: just select best clustering
- No need to manually engineer distance metric
- Faster, better final clustering for task at hand

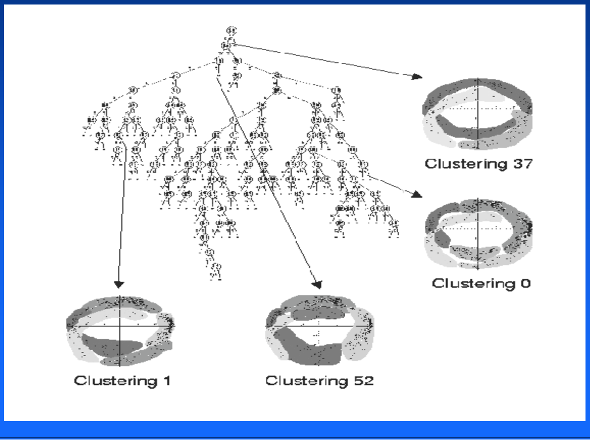
Main Goals

- Move away from clustering at the "assembly language level"
- Push as much work as possible required for clustering from the user to the computer
- Make clustering as automatic as possible
- More effective clustering in hands of users, not researchers
- Find better clusters/clusterings
- Find better clusters/clusterings faster
- Simultaneously provide multiple/alternate views of data
- Meta level helps users understand complex data faster
- Provide more natural user controls and feedback

Multiple Different Clusterings



MetaCluster: Cluster of *Clusterings*



Research Questions

- How to generate different clusterings?
  - Automatically adjust distance metric
  - EM/k-means gets stuck in local minima
  - Stochastic clustering (bagging HAC)
- How to measure distance between clusterings?
  - Hop distance (hierarchical clustering methods)
  - Pairwise overlap
- How to organize clusterings for user?
  - Hierarchical MDS?
- How to combine/merge clusterings?

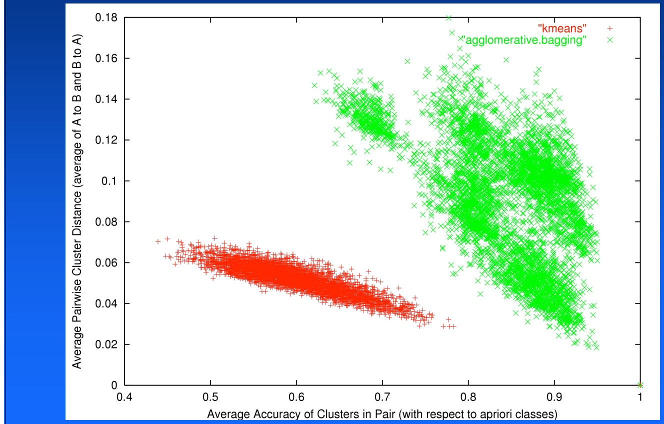
Q1: Generating Alternate Clusterings

- EM/k-means clustering
  - Iterated random restarts
- HAC: hierarchical agglomerative clustering
  - Stochastic bagged agglomerative clustering
- MDS: multidimensional scaling
  - Eigenvector analysis
  - Feature creation, selection, and weighting
- Automatic distance metric learning
  - Find distance metric that responds to user feedback

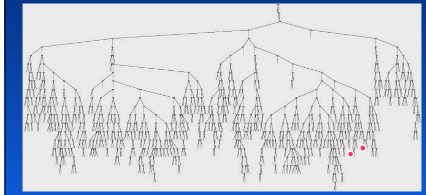
Meta Clustering Text Documents

- focused web crawls in 26 categories
- 30,000 web documents
- bags-of-words distance between documents
- cluster documents
  - k-means
  - hierarchical clustering
- calculate hop-distance between clusterings
- cluster clusterings

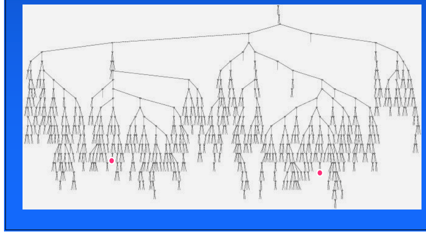
Meta Clustering Text Documents



Q2: Comparing Entire Clusterings



$\text{dist}_{c1}(i,j) = 7$

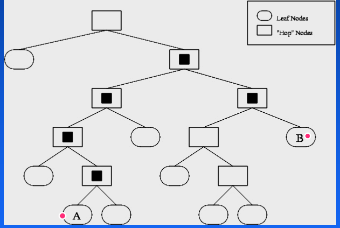


$\text{dist}_{c2}(i,j) = 30$

$\sum \ldots (7-30)^{2\ldots}$

Hop Distance

Compute distance between clusterings pairwise:

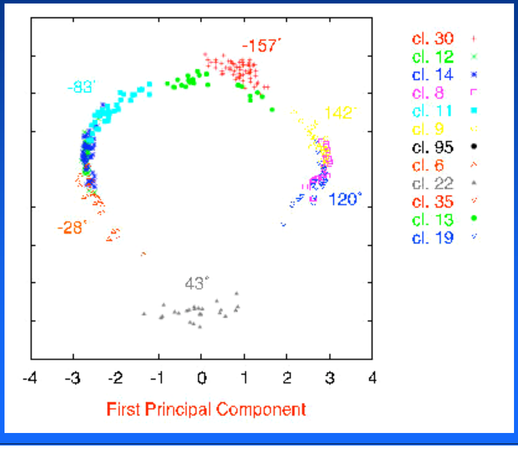


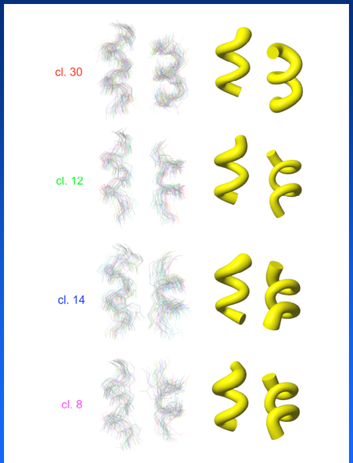
$$\sum_{i,j}\left(HOP_{c1}(i,j) - HOP_{c2}(i,j)\right)^{\alpha}$$

Q3: Understanding Clusters

- Visualization
- Textual summarization
- Multiple parallel views
- Easy group summarization
- Easy point navigation
- Translate to DB query or distance metric

Q4: Cluster Visualization/Interpretation





Research Questions

- How to generate qualitatively different clusterings?
- How to measure similarity between entire clusterings?
- How best to cluster the clusterings at the meta level?
- How to aid cluster/clustering visualization, interpretation, and summarization?
- How to provide and respond to user feedback?
- How to convert best clustering to a distance metric that can be used for other purposes beyond the clustering?
- How to combine meta clustering with other data analysis?
- How to evaluate clustering performance?

Research

- Collaboration with real users
  - Not "textbook" clustering
  - Real data
  - Real needs
- Computational power to generate clusterings
- Real Experts
  - Interpret clusterings
  - Feedback on overall system
  - Measure usefulness
- Benchmark problems

Ultimately, users want a clustering system that can be driven like a car, with a set of user controls, a dashboard full of indicators, and a windshield on the current clustering

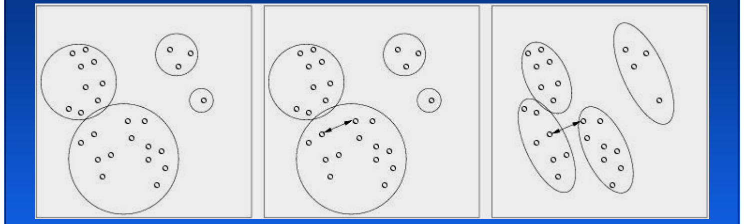
Text Clustering With User Feedback

Yahoo! Problem

- 100,000 documents (papers, web pages, articles)
- Group them into classes or a hierarchy
- Not told what classes or hierarchy to use
- Group so docs can be browsed/retreived easily
- Have criteria in mind, but can’t verbalize them
- Yahoo! Problem is ubiquitous

- There is no such thing as the ***right*** clustering, but...
- Like art, you know good/bad clusters when you see them

It’s Easier to Criticize Than to Create



- Do initial unsupervised clustering
- “Browse” clusters
- Criticize clustering
- Re-cluster with algorithm sensitive to your critique
- Repeat until happy with final clustering

Types of Critiques/Constraints

- This pt doesn’t belong here
- Move this pt to that cluster
- These two pts shouldn’t be in the same cluster
- These two pts should be in the same cluster
- This cluster sucks
- This cluster is good
- This cluster is too small/large
- The label for this pt is “X”
- Move this cluster near that one
- Move this cluster up/down towards/away from the root
- The clustering is too coarse
- Give me a different clustering
- Give me a clustering half way between these two clusterings
- Cluster using these features
- Cluster using this method
- ...

Constraints ≠ Labels

- Constraints weaker than labels
  - Don’t have to know labels of pts to know that pts should or should not be grouped together
  - Don’t even have to know what the legal labels are
  - More pairwise constraints between pts than labelings
  - Users can generate constraints more easily than labels

Text Experiment 1

- 20,000 USENET articles
- Four subjects (we know labels, clusterer does not):

Aviation Simulators	Real Aviation
Automobile Simulators	Real Automobiles

- Different users give different pairwise constraints:

Results

- 50% accuracy for unsupervised clustering
- Add 10 pairwise constraints (20,000 articles)
- 80% accuracy if constraints used to warp distance metric for unsupervised clustering

Text Experiment 2

Compare
Supervised Learning with Labels to
Semi-Supervised Clustering with User Constraints

Naive Bayes Bags-of-Words Model

Generative Model:

$$p(d) = \prod_{w_j \in V} p(w_j \mid \theta)^{N(w_j, d)}$$

Natural Distance Metric:

$$KLD_M(d_1 \| d_2) = |d_1| KLD(d_1 \| M_{12}) + |d_2| KLD(d_2 \| M_{12})$$

To Implement Constraints

Modify KL-Divergence:

$$KLD' = \prod_{w_j \in V} \lambda_j \cdot p(w_j \mid \theta_{d1}) \log\left(\frac{p(w_j \mid \theta_{d2})}{p(w_j \mid \theta_{d1})}\right)$$

where the lambda weights control the warping of the distance metric for each word.

Adjusting the Weights

Given constraint that documents d_1 and d_2 should be in different clusters:

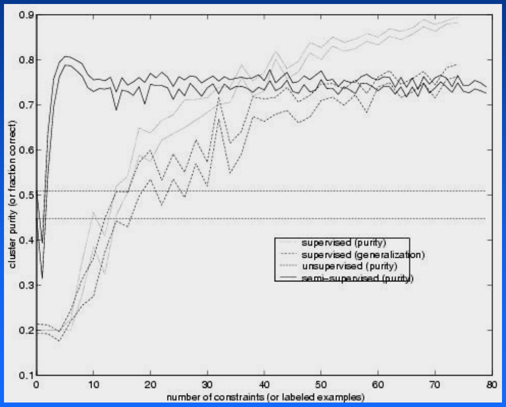
$$\frac{\partial\, KLD'_M(d_1 \parallel d_2)}{\partial \lambda_j} = |d_1| p(w_j \mid \theta_{d1}) \log\left(\frac{p(w_j \mid \theta_{d1d2})}{p(w_j \mid \theta_{d1})}\right) +$$
$$|d_2| p(w_j \mid \theta_{d2}) \log\left(\frac{p(w_j \mid \theta_{d1d2})}{p(w_j \mid \theta_{d2})}\right)$$

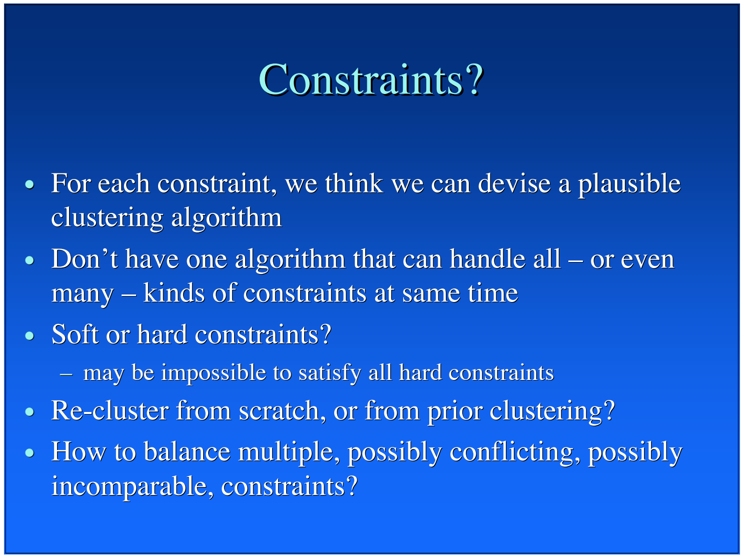
Experiment

- 5 Reuters topic areas:

  business, health, politics, sports, tech
- 25 documents each
- EM-based clustering:
  - from p(d|c) compute p(c|d)
  - Soft clustering based on p(c|d)
  - Update cluster parameters
- Add constraints one-at-a-time that documents with different labels should not be in same cluster

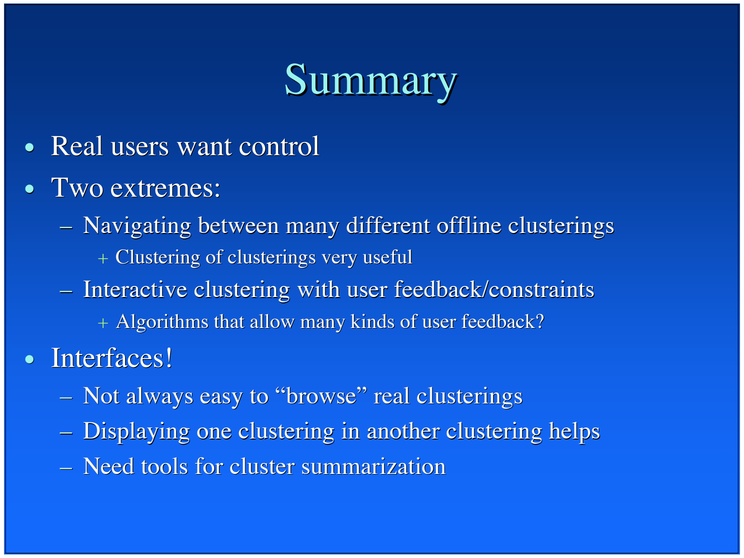
Results





Constraints?

- For each constraint, we think we can devise a plausible clustering algorithm
- Don’t have one algorithm that can handle all – or even many – kinds of constraints at same time
- Soft or hard constraints?
  - may be impossible to satisfy all hard constraints
- Re-cluster from scratch, or from prior clustering?
- How to balance multiple, possibly conflicting, possibly incomparable, constraints?



Summary

- Real users want control
- Two extremes:
  - Navigating between many different offline clusterings
    - Clustering of clusterings very useful
  - Interactive clustering with user feedback/constraints
    - Algorithms that allow many kinds of user feedback?
- Interfaces!
  - Not always easy to “browse” real clusterings
  - Displaying one clustering in another clustering helps
  - Need tools for cluster summarization