Information Retrieval

INFO 4300 / CS 4300

- Presenting Results
  - Clustering

Clustering Results

- Result lists often contain documents related to different *aspects* of the query topic
- ***Clustering*** is used to group related documents to simplify browsing

Example clusters for query "tropical fish"

Pictures (38)
Aquarium Fish (28)
Tropical Fish Aquarium (26)
Exporter (31)
Supplies (32)
Plants, Aquatic (18)
Fish Tank (15)
Breeding (16)
Marine Fish (16)
Aquaria (9)

Result List Example

1. **Badmans Tropical Fish**
   A freshwater aquarium page covering all aspects of the **tropical fish** hobby. ... to Badman's **Tropical Fish**. ... world of aquariology with Badman's **Tropical Fish**. ...
2. **Tropical Fish**
   Notes on a few species and a gallery of photos of African cichlids.
3. The **Tropical** Tank Homepage - **Tropical Fish** and Aquariums
   Info on **tropical fish** and **tropical** aquariums, large **fish** species index with ... Here you will find lots of information on **Tropical Fish** and Aquariums. ...
4. **Tropical Fish** Centre
   Offers a range of aquarium products, advice on choosing species, feeding, and health care, and a discussion board.
5. **Tropical fish** - Wikipedia, the free encyclopedia
   Tropical fish are popular aquarium **fish** , due to their often bright coloration. ... Practical Fishkeeping • **Tropical Fish** Hobbyist • Koi. Aquarium related companies: ...
6. **Tropical Fish** Find
   Home page for **Tropical Fish** Internet Directory ... stores, forums, clubs, **fish** facts, **tropical fish** compatibility and aquarium ...
7. Breeding **tropical fish**
   ... intrested in keeping and/or breeding **Tropical**, Marine, Pond and Coldwater **fish**. ... Breeding **Tropical Fish** ... breeding **tropical**, marine, coldwater & pond **fish**. ...
8. FishLore
   Includes **tropical** freshwater aquarium how-to guides, FAQs, **fish** profiles, articles, and forums.
9. Cathy's **Tropical Fish** Keeping
   Information on setting up and maintaining a successful freshwater aquarium.
10. **Tropical Fish** Place
    Tropical Fish information for your freshwater **fish** tank ... great amount of information about a great hobby, a freshwater **tropical fish** tank. ...

Top 10 documents for "tropical fish"

Clustering Results

- Requirements
- Efficiency (NP-hard)
  - generated online, i.e. in real time
  - must be specific to each query and are based on the top-ranked documents for that query
  - typically based on snippets
- Easy to understand
  - Can be difficult to assign good labels to groups
  - Monothetic vs. polythetic classification

Types of Classification

- Monothetic
  - every member of a class has the property that defines the class
  - typical assumption made by users
  - easy to understand
- Polythetic
  - members of classes share many properties but there is no single defining property
  - most clustering algorithms (e.g. K-means) produce this type of output

Classification Example

$$D_1 = \{a, b, c\}$$
$$D_2 = \{a, d, e\}$$
$$D_3 = \{d, e, f, g\}$$
$$D_4 = \{f, g\}$$

- Possible monothetic classification
  - $\{D_1, D_2\}$ (labeled using *a*) and $\{D_2, D_3\}$ (labeled *e*)
- Possible polythetic classification
  - $\{D_2, D_3, D_4\}$, D_1
  - labels?

Result Clusters

- Simple algorithm
  - group based on snippet non-stopwords

aquarium (5)	(1, 3, 4, 5, 8)
freshwater (4)	(1, 8, 9, 10)
species (3)	(2, 3, 4)
hobby (3)	(1, 5, 10)
forums (2)	(6, 8)

- Refinements
  - use phrases
  - use more features
    - » whether phrases occurred in titles or snippets
    - » length of the phrase
    - » collection frequency of the phrase
    - » overlap of the resulting clusters

Classification and Clustering

- Classification and clustering are classical pattern recognition / machine learning problems
- Classification
  - Asks “what class does this item belong to?”
  - *Supervised learning* task
- Clustering
  - Asks “how can I group this set of items?”
  - *Unsupervised learning* task
- Items can be documents, queries, emails, entities, images, etc.
- Useful for a wide variety of search engine tasks

Classification

- Classification is the task of automatically applying labels to items
- Useful for many search-related tasks
  - Spam detection
  - Sentiment classification
  - Online advertising
  - **Identifying fake online reviews**
- Two common approaches
  - Probabilistic
  - Geometric

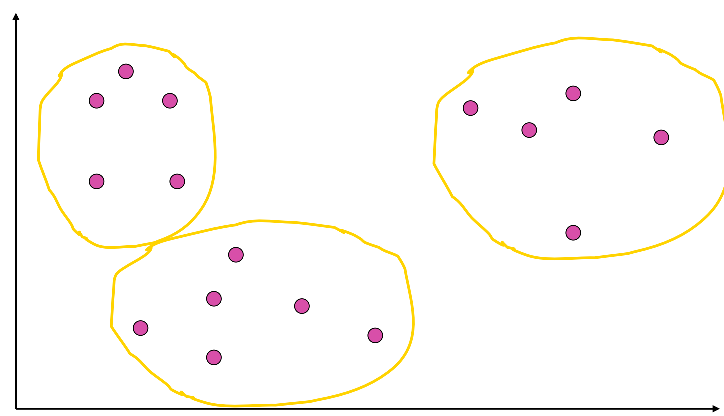
Clustering

- A set of unsupervised algorithms that attempt to find latent structure in a set of items
- Goal is to identify groups (clusters) of similar items

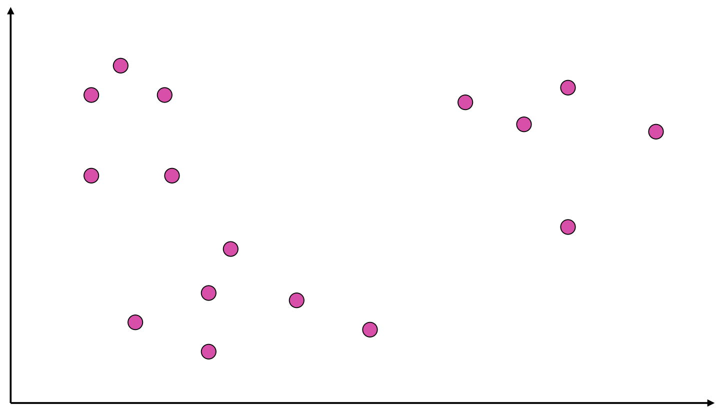
Clustering

- General outline of clustering algorithms
  1. Decide how items will be represented (e.g., feature vectors)
  2. Define similarity measure between pairs or groups of items (e.g., cosine similarity)
  3. Determine what makes a “good” clustering
  4. Iteratively construct clusters that are increasingly “good”
  5. Stop after a local/global optimum clustering is found
- Steps 3 and 4 differ the most across algorithms

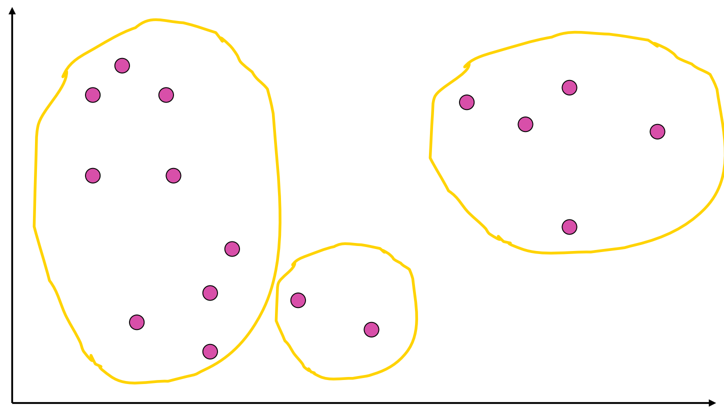
Clustering example



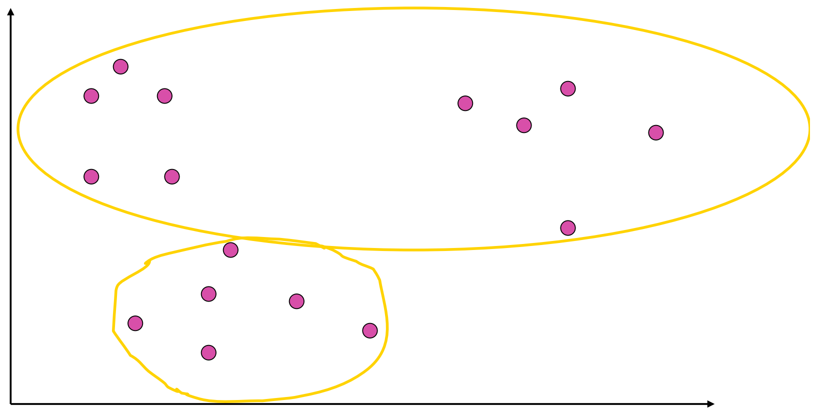
Clustering example



Clustering example



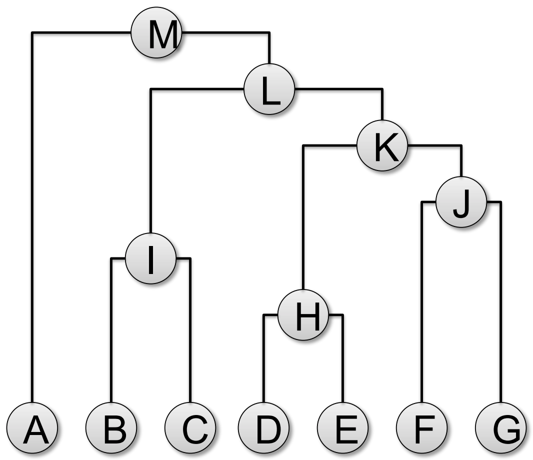
Clustering example



Hierarchical Clustering

- Constructs a hierarchy of clusters
  - The top level of the hierarchy consists of a single cluster with all items in it
  - The bottom level of the hierarchy consists of *N* (# items) singleton clusters
- Two types of hierarchical clustering
  - Divisive ("top down")
  - Agglomerative ("bottom up")
- Hierarchy can be visualized as a *dendogram*

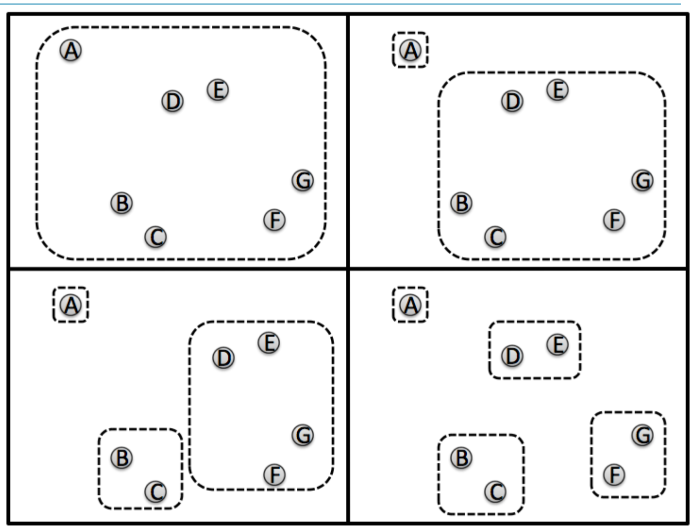
Example Dendrogram



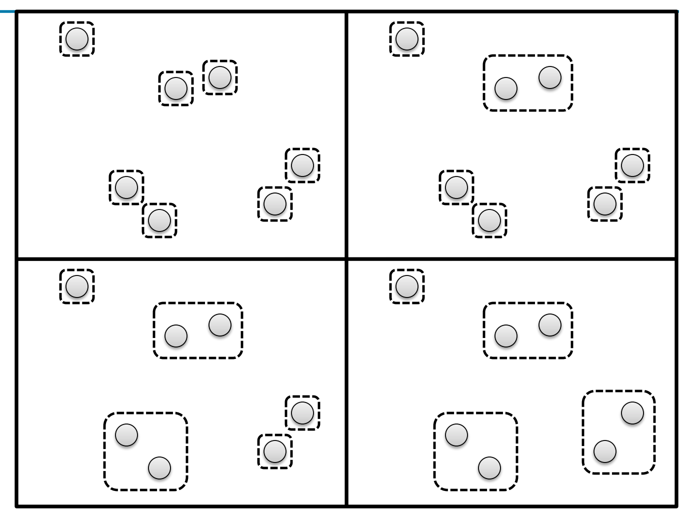
Divisive and Agglomerative Hierarchical Clustering

- Divisive
  - Start with a single cluster consisting of all of the items
  - Until only singleton clusters exist…
    - » **Divide** an existing cluster into two new clusters
- Agglomerative
  - Start with *N* (# items) singleton clusters
  - Until a single cluster exists…
    - » **Combine** two existing cluster into a new cluster
- How do we know how to divide or combined clusters?
  - Define a division or combination cost
  - Perform the division or combination with the lowest cost

Divisive Hierarchical Clustering



Agglomerative Hierarchical Clustering



Agglomerative Clustering (HAC)

Algorithm 1 Agglomerative Clustering

```
procedure AgglomerativeCluster(X_1, ..., X_N, K)
    A[1], ..., A[N] ← 1, ..., N
    ids ← {1, ..., N}
    for c = N to K do
        bestcost ← ∞
        bestclusterA ← undefined
        bestclusterB ← undefined
        for i ∈ ids do
            for j ∈ ids − {i} do
                c_{i,j} ← COST(C_i, C_j)
                if c_{i,j} < bestcost then
                    bestcost ← c_{i,j}
                    bestclusterA ← i
                    bestclusterB ← j
                end if
            end for
        end for
        ids ← ids − {bestClusterA}
        for i = 1 to N do
            if A[i] is equal to bestClusterA then
                A[i] ← bestClusterB
            end if
        end for
    end for
end procedure
```

Clustering Costs

- Single linkage

$$COST(C_i, C_j) = \min\{dist(X_i, X_j) | X_i \in C_i, X_j \in C_j\}$$

- Complete linkage

$$COST(C_i, C_j) = \max\{dist(X_i, X_j) | X_i \in C_i, X_j \in C_j\}$$

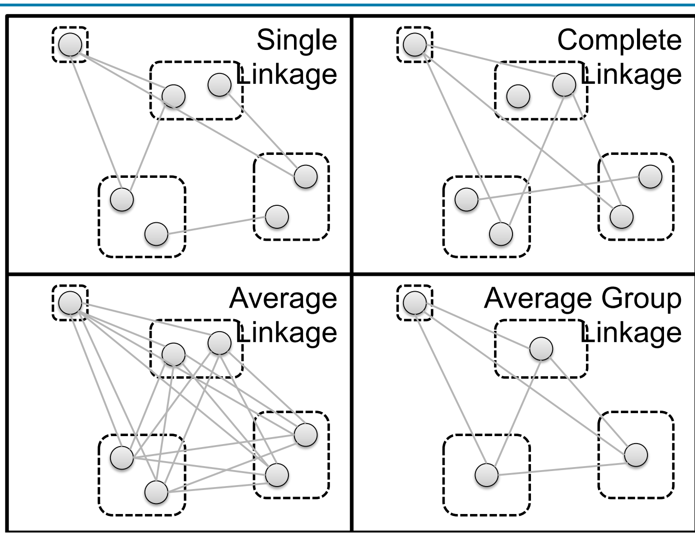
- Average linkage

$$COST(C_i, C_j) = \frac{\sum_{X_i \in C_i, X_j \in C_j} dist(X_i, X_j)}{|C_i||C_j|}$$

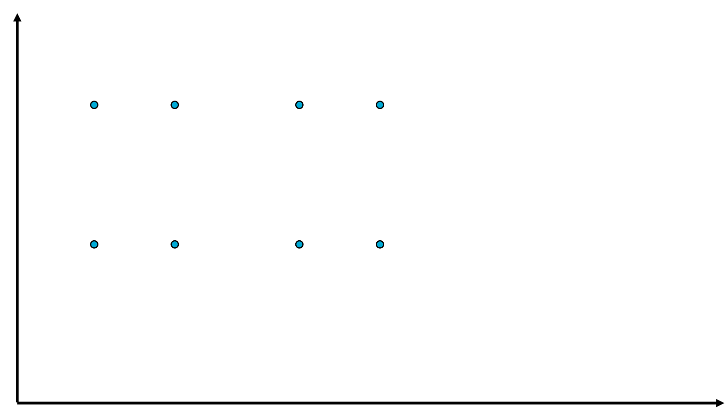
- Average group linkage

$$COST(C_i, C_j) = dist(\mu_{C_i}, \mu_{C_j})$$

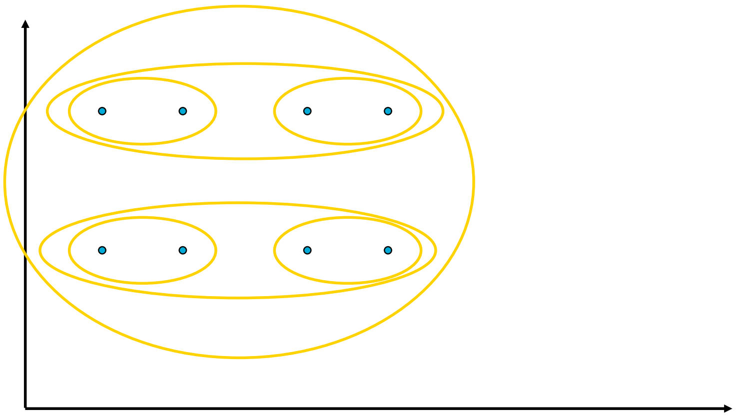
Clustering Strategies



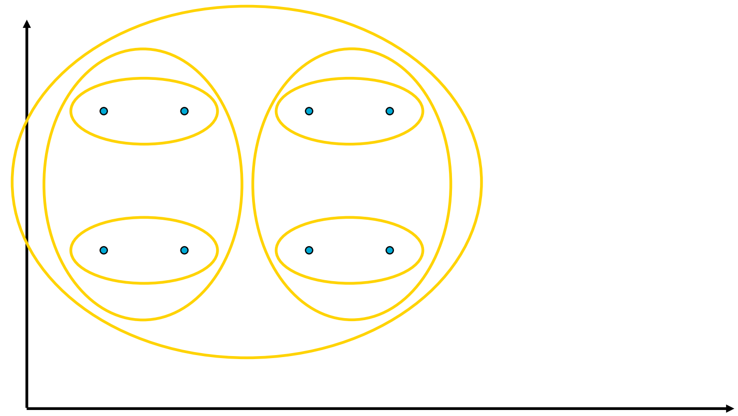
Single link example



Single link example



Complete link example



Computational complexity of HAC

- In the first iteration, all HAC methods need to compute similarity of all pairs of n individual instances which is $O(n^2)$.
- In each of the subsequent n-2 merging iterations, it must compute the distance between the most recently created cluster and all other existing clusters.
- In order to maintain an overall $O(n^2)$ performance, computing similarity to each other cluster must be done in constant time.

K-Means Clustering

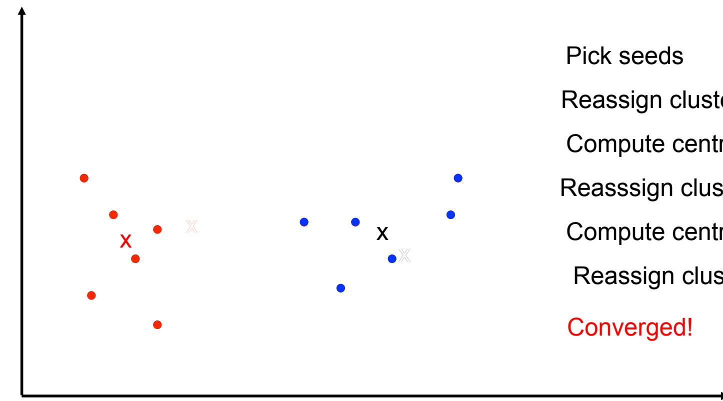
- Hierarchical clustering constructs a hierarchy of clusters
- K-means always maintains exactly K clusters
  - Clusters represented as centroids ("center of mass")
- Basic algorithm:
  - Step 0: Choose K cluster centroids
  - Step 1: Assign points to closest centroid
  - Step 2: Recompute cluster centroids
  - Step 3: Goto 1
- Tends to converge quickly
- Can be sensitive to choice of initial centroids
- Must choose K!

K-Means Clustering Algorithm

Algorithm 1 K-Means Clustering

procedure KMEANSCLUSTER($X_1, \ldots, X_N$, K)
  $A[1], \ldots, A[N] \leftarrow$ initial cluster assignment
  repeat
    $change \leftarrow false$
    for $i = 1$ to N **do**
      $\hat{k} \leftarrow \arg\min_k dist(X_i, C_k)$
      if $A[i]$ **is not equal** $\hat{k}$ **then**
        $A[i] \leftarrow \hat{k}$
        $change \leftarrow true$
      end if
    end for
  until $change$ **is equal to** $false$ **return** $A[1], \ldots, A[N]$
end procedure

K-means example (k=2)



Pick seeds

Reassign clusters

Compute centroids

Reasssign clusters

Compute centroids

Reassign clusters

Converged!

Time complexity

- Assume computing distance between two instances is O(*m*) where *m* is the dimensionality of the vectors.
- Reassigning clusters for *n* points: O(*kn*) distance computations, or O(*knm*).
- Computing centroids: Each instance gets added once to some centroid: O(*nm*).
- Assume these two steps are each done once for *i* iterations: O(*iknm*).
- Linear in all relevant factors, assuming a fixed number of iterations, more efficient than O(n^2) HAC.

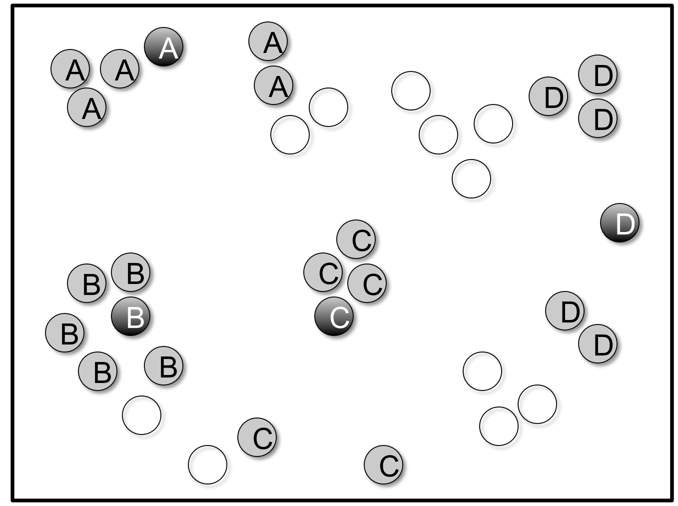
Seed choice

- Results can vary based on random seed selection.
- Some seeds can result in poor convergence rate, or convergence to sub-optimal clusterings.
- Select good seeds using a heuristic or the results of another method.

K-Nearest Neighbor Clustering

- Hierarchical and K-Means clustering partition items into clusters
  - Every item is in exactly one cluster
- K-Nearest neighbor clustering forms one cluster per item
  - The cluster for item *j* consists of *j* and *j*'s *K* nearest neighbors
  - Clusters now overlap

5-Nearest Neighbor Clustering



Evaluating Clustering

- Evaluating clustering is challenging, since it is an ***unsupervised*** learning task
- If labels exist, can use standard IR metrics, such as precision and recall
- If not, then can use measures such as "cluster precision", which is defined as:

$$ClusterPrecision = \frac{\sum_{i=1}^{K} |\text{MaxClass}(C_i)|}{N}$$

- Another option is to evaluate clustering as part of an end-to-end system

How to Choose K?

- K-means and K-nearest neighbor clustering require us to choose *K*, the number of clusters
- No theoretically appealing way of choosing *K*
- Depends on the application and data
- Can use hierarchical clustering and choose the best level of the hierarchy to use
- Can use adaptive *K* for K-nearest neighbor clustering
  - Define a 'ball' around each item
- Difficult problem with no clear solution

Applications of document clustering

- Cluster retrieved documents
  - to present more organized and understandable results to user
- Cluster documents in collection (global analysis)
  - during retrieval, add other documents in the same cluster as the initial retrieved documents to improve recall
- Automated (or semi-automated) creation of document taxonomies
  - e.g. Yahoo-style
- Improve document representation
  - e.g. probabilistic LSI [Hofmann SIGIR 98]

Applications of document clustering

- Cluster-based browsing: scatter/gather

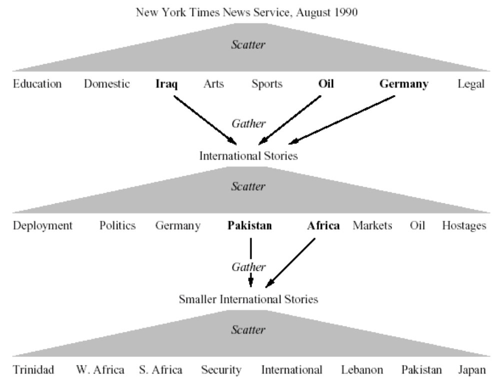


Figure 1: Illustration of Scatter/Gather