

# Scalable Recursive Top-Down Hierarchical Clustering Approach with implicit Model Selection for Textual Data Sets






Markus Muhr, Vedran Sabol, Michael  
Granitzer

Know-Center GmbH & Graz University of  
Technology




<http://www.know-center.at>

# Outline


## Motivation

-  Facetted Retrieval
-  Scatter/Gather
-  Visual Analysis of unstructured document sets

## Clustering Approach

-  Overview
-  Growing k-means
-  Modifications

## Experiments

-  Visual Analysis
-  Inex

# Motivation

# Facetted Retrieval

The screenshot shows a search engine interface for the query "clustering". The main search bar contains "clustering" and a search button. Below the search bar, there are navigation tabs for "clouds", "sources", and "sites". The "clouds" tab is active, showing a list of facets with counts:

- All Results (236)
- Search (27)
- Clustering Engine (9)
- Database (6)
- Works together, Search Engines (2)
- Windows Server (3)
- Other Topics (7)
- Hierarchical (15)
- Linux (14)
- Computing (18)
- Algorithm (19)
- Modeling (11)
- High availability (17)
- Management (9)
- Dimensional (8)
- Sells (7)

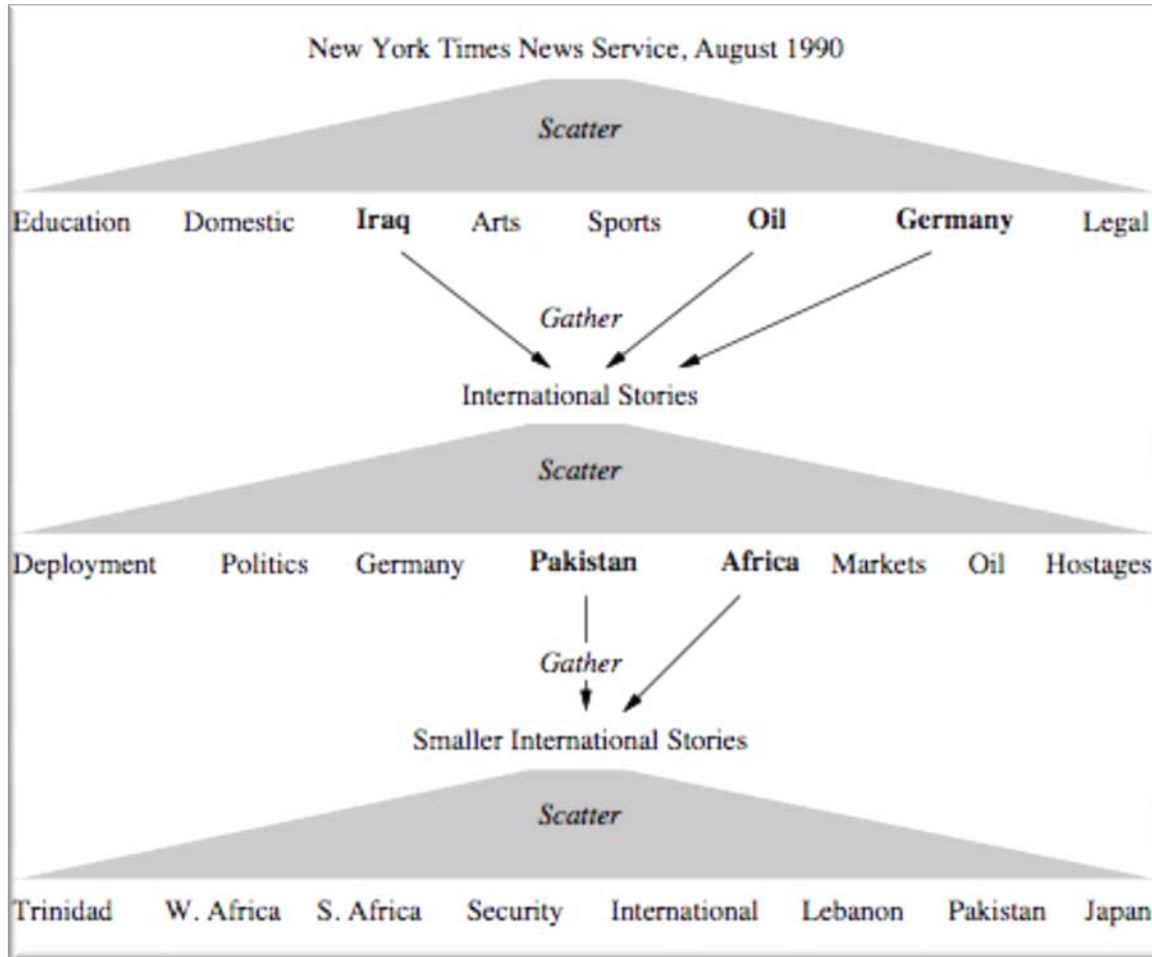
Below the facets, there are links for "more" and "all clouds". To the right of the facets, a list of search results is displayed under the heading "Cluster Search contains 27 documents.":

- Vivisimo Clustering Engine** - Their document clustering and meta-search software automatic vivisimo.com - [cache] - Open Directory, Ask
- XSearch** - Groups results into meaningful groups by clustering technology. www.xclustering.com - [cache] - Open Directory, Ask
- PolyMeta** - Intelligent general meta search and clustering engine. Custom e www.polymeta.com - [cache] - Open Directory
- SnakeT** - A metasearch clustering engine for Web, Books, News. snaket.di.unipi.it - [cache] - Open Directory
- ToxSeek** - Metasearch and Clustering Engine for Environmental Health and toxseek.nlm.nih.gov - [cache] - Open Directory
- Open Text Federated Query Server** - Web search to demonstrate how this search solution can unify a queryserver.opentext.com/web.htm - [cache] - Open Directory
- AllPlus** - Universal metasearch and discovery engine with clustering. Dis

The interface also includes a sidebar with navigation options like "Everything", "Images", "Videos", "Maps", "News", "Shopping", "Books", "Blogs", "Updates", "Discussions", and "Fewer". There are also filters for "Any time" (Latest, Past 24 hours, Past week, Past month, Past year, Custom range...) and "All results" (Social, Nearby, Visited pages, Not yet visited).

# Motivation

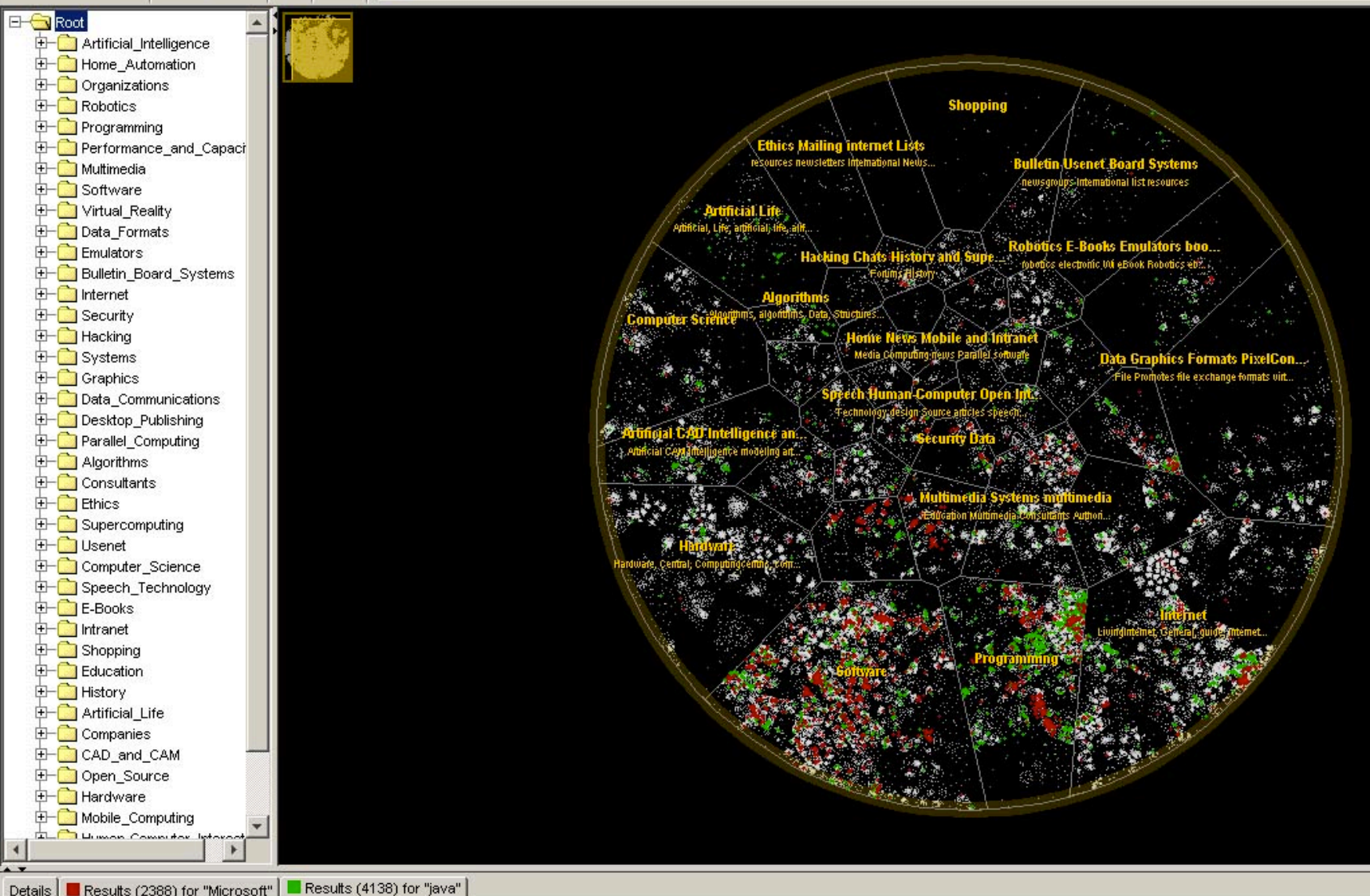
## Scatter/Gather [Cutting et. al. 1992]





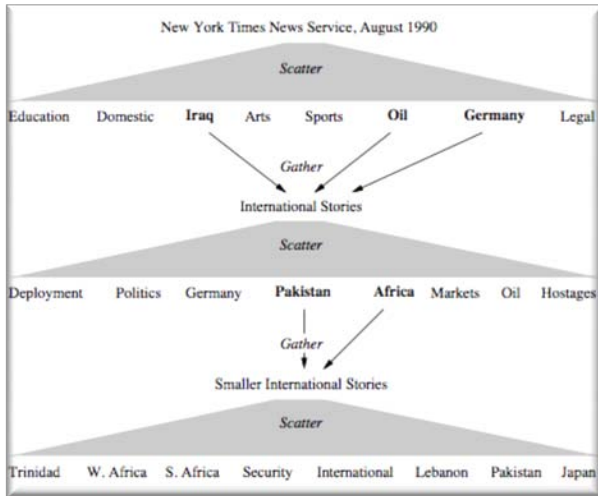
# Motivation

# InfoSky: Visual Exploration [Andrews et. al. 2002]

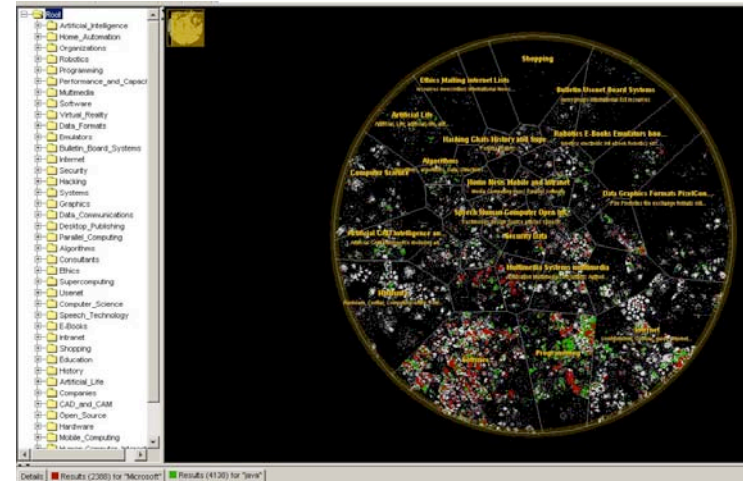


# Motivation

## InfoSky + Scatter/Gather



+



- Automatic creation of the cluster hierarchy while retaining InfoSky's analysis capabilities

### Questions

- What is an efficient hierarchical clustering algorithm therefore?
- How to combine statistical data set properties with visual requirements?



# Clustering Contributions

- Hierarchical, top-down, polythetic, document clustering approach
- Dynamic cluster structure on each level of the hierarchy supporting splitting and merging of clusters.
- Constraints on the maximum and minimum number of elements per hierarchy level
- Resulting reduced computational costs of the layout algorithm
- Scalable to datasets consisting of millions of documents with a reasonable trade-off between runtime and accuracy



**Top-Down, scalable clustering algorithm for creating a topical hierarchy**



# Clustering Overview

Divide and conquer: decompose into tasks starting at the root node

For every task

- Step 1: Preprocess documents to be clustered
  - Bag-of-Words, BM 25, cosine inner product
- Step 2: Cluster documents using a flat clustering algorithm
- Step 3: Split and merge clusters till constraints are met
- Step 4: Recursion: Evaluate the stopping criterion for dividing into further sub-tasks
- Step 5: Cluster Labeling
- Step 6: Project clusters into a 2 dimensional space





# Clustering

## Step 2: Clustering Algorithm (1/4)

Given a set of documents  $X$ , find a set of  $K$  groups of similar documents (clusters)

- Utilize existing clustering methods

HAC, DBScan or Chameleon  $> O(n^2)$

BIRCH fast and storage efficient, but order dependent

- Growing k-means -

Online Competitive Learning with Winner-takes it all approach

trade-off between runtime and accuracy [Zhao and Karypis 02]

Allows for efficient model selection (determine  $k$ )

# Clustering

## Step 2: Clustering Algorithm (2/4)

---

### Algorithm 1 Growing Spherical K-Means

---

**input:**

$\mathcal{X} = \{x_1, \dots, x_N\}$  with  $x_i \in \mathbb{R}^d$ ,  $K, l, \eta, \nu$

**output:**

$\mathcal{C} = \{c_1, \dots, c_K\}$ ,  $\mathcal{Y} = \{y_1, \dots, y_N\} \forall y_n \in \{1, \dots, K\}$

**steps:**

initialize centroids  $c_1$  and  $c_2$  by a seeding mechanism

**for**  $m = 2$  to  $K$  **do**

**for**  $n = 1$  to  $N$  **do**

$y_p = y_n$

$y_n = \arg \max_{1 \leq k \leq m} x_n^T c_k$

$c_{y_n} = c_{y_n} + \eta x_n$

$c_{y_p} = c_{y_p} - \nu x_n$

**if**  $\|c_{y_n}\| - 1.0 > l$  **then**

$c_{y_n} = \frac{c_{y_n}}{\|c_{y_n}\|}$

**for**  $n = 1$  to  $N$  **do**

$y_n = \arg \max_{1 \leq k \leq m} x_n^T c_k$

$s_k = s_k + \max_{1 \leq k \leq m} x_n^T c_k$

**if**  $m < K$  **then**

$c_i = \arg \min_{1 \leq k \leq m} S(c_k)$

$x_j = \arg \min_{x \in \mathcal{X}_i} x^T c_i$  with  $\mathcal{X}_i = \{x_n | y_n = i\}$

$c_t = \frac{c_i - x_j}{2}$ ,  $\mathcal{C} = \mathcal{C} \cup \{c_t\}$

Init and loopformaximumk-clusters

Update clusterhypothesis

Runtimeimprovement of centroid update

Assigndocuments andaveragesimilarity

Createm-thcentroid

# Clustering

## Step 2: Clustering Algorithm (3/4)

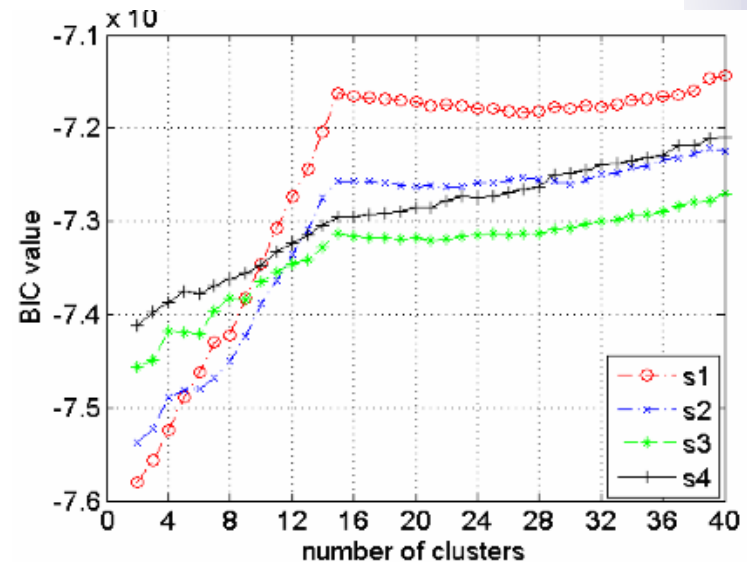
### Model Selection methods

Obtain fitness criterion for different number of clusters (Bayesian Information Criterion (BIC), Stability based approaches)

Monotonical increasing/decreasing

Overtraining on the data

Determine the „best cluster number“  
using knee-point detection  
[Zhao et. al. 2008]



Efficient calculation for the growing k-means by simply calculating the fitness criterion for each new centroid

# Clustering

## Step 2: Clustering Algorithm (4/4)

### Heuristics

- Efficient update rules [Zhong 2005]

Move a fraction of the distance  
between sample and centroid

$$c_{y_n} = \frac{c_{y_n} + \eta(x_n - c_{y_n})}{\|c_{y_n} + \eta(x_n - c_{y_n})\|}$$

Simply update the angle and ignore  
non unit length

Track norm changes and rescale after  
norm exceeds numerical boundaries

$$c_{y_n} = c_{y_n} + \eta x_n$$

$$c_{y_p} = c_{y_p} - \nu x_n$$

**if**  $\|c_{y_n}\| - 1.0 > l$  **then**

$$c_{y_n} = \frac{c_{y_n}}{\|c_{y_n}\|}$$

- Decreasing learning rate with the size of the cluster for balancing

$$\eta = 1 / |\sqrt{\mathcal{X}_{k(x)}}|$$



# Clustering

## Step 3: Split and Merge

Split and Merge Clusters to fulfill the following constraints

- # Cluster at one level

Merge the most similar cluster if  $\#cluster > \text{maximum number of clusters}$

Split the least coherent or biggest cluster if  $\#cluster < \text{minimum number of clusters}$

- # documents in a cluster

Below the Maximum number of documents for a cluster →  
`clusterokforbrowsing`

More than 1.5 times the upper limit to ensure meaningful clustering at next hierarchical level

If all clusters fulfill this constraint, cluster recursively  
(Step 4)





# Clustering

## Step 5&6: Labeling & Projection

- Labeling via Jensen Shannon Divergence

  - JSD best suited

  - Exploit hierarchical structures (not focus of this work)

- Projection [Andrews et. Al. 2004]

  - Force directed placement  $O(n^3)$

  - Recursive application on cluster hierarchy using document and cluster centroids as points to layout

  - Due to the constraints we achieve a runtime of roughly  $O(n \cdot \log(n))$

  - Voronoi inscription of rectangular Layout





# Experiments

## Clustering based Visualisation

- Preliminary user evaluation

  - Combination of visualisation and standard components helpful for explorative tasks [Andrews et. Al. 2002]

  - Improved interaction and navigation paradigms to support explorative search tasks

  - Patent analysis tasks improved in real world use case

  - Suitable for high recall search tasks

- Detailed evaluation still missing.



# Experiments

## INEX Clustering

- Initiativ for Evaluation of XML Retrieval
- XML Mining Track – Cluster the English Wikipedia
  - Small data set 54k documents
  - Large data set 2.6 Million Documents
  - Preprocessed document vectors (uni and bi-grams)
- Ground truth provided by YAGO ontology, but no hierarchical structure
- Document assigned to each cluster on the path to facilitat multi cluster assignment as it is the case in Wikipedia



# Experiments

## INEX Clustering

- 10,467 Clusters for the small data set

4 Minutes to compute on a 16GB Quad Core including I/O

MacroPurity	BIC	Stability
73k Categories	0.4959	0.4945
12k Categories	0.5473	0.5303

- 133,704 Clusters on the large data set

Runtime 2 hours

348 k Categories: Macro Purity of 0.4457

12k Categories: Macro Purity of 0.5359

- Clusters appear to be reasonable, but good evaluation strategy remains an open issue

High level clusters are more important

Accurate ground truth reflecting good browsing strategies





# Summary & Conclusio

- Motivation: Support explorative search tasks via Retrieval by browsing
- Needed: Scalable Clustering algorithm
  - Hierarchie Layout as constraint
  - Model selection
- Top-down, recursive algorithm with different model selection strategy
- Experiments
  - Used in visual analysis application
  - INEX Clustering evaluation
- Evaluation for explorative analysis task remains an open problem

Thank for your attention  
**Questions?**



**Michael Granitzer**

Scientific Director  
Know-Center Graz  
Inffeldgasse 21a  
8020 Graz

+43 316 873 9263  
[mgrani@know-center.at](mailto:mgrani@know-center.at)  
[www.know-center.at](http://www.know-center.at)