Canopy Clustering

Oresti Konomi
Michail Romanovic Melnikov

Fachgebiet Datenbanksysteme und Informationsmanagement
Technische Universität Berlin

http://www.dima.tu-berlin.de/

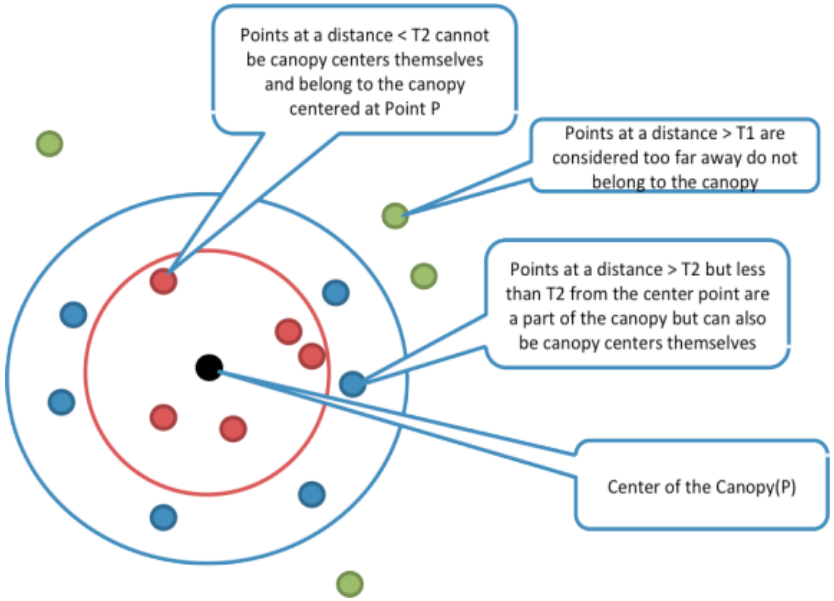
Overview

Canopy Clustering

- Introduction
- Example
- Characteristics
- PACT Plan
- Benchmarks

Canopy Clustering:

- Intended to speed up clustering on large data sets
  - Large number of elements
  - Large number of features
  - Large number of clusters
- Often used as preprocessing step for clustering algorithm like K-Means
- Preliminary step to help parallelize computation.
- Clusters data into overlapping canopies using super cheap distance metric
- Efficient, Accurate



Algorithm

Given two threshold distance T1 and T2 with T1>T2 and a set of points.

1. Determine Canopy Centers
   - Iterate through the set of points, if the point is at distance determine the canopy membership:
     - for each point in the input set if the point is at a distance < T1 from any of points In the list of canopy centers (generated in step1) then point is member of the corresponding canopy.
2. …

The algorithm proceeds as follows:

- Cheaply partitioning the data into overlapping subsets (called 'canopies')
- Perform more expensive clustering but only within these canopies.

Example Algorithm

K-Means:

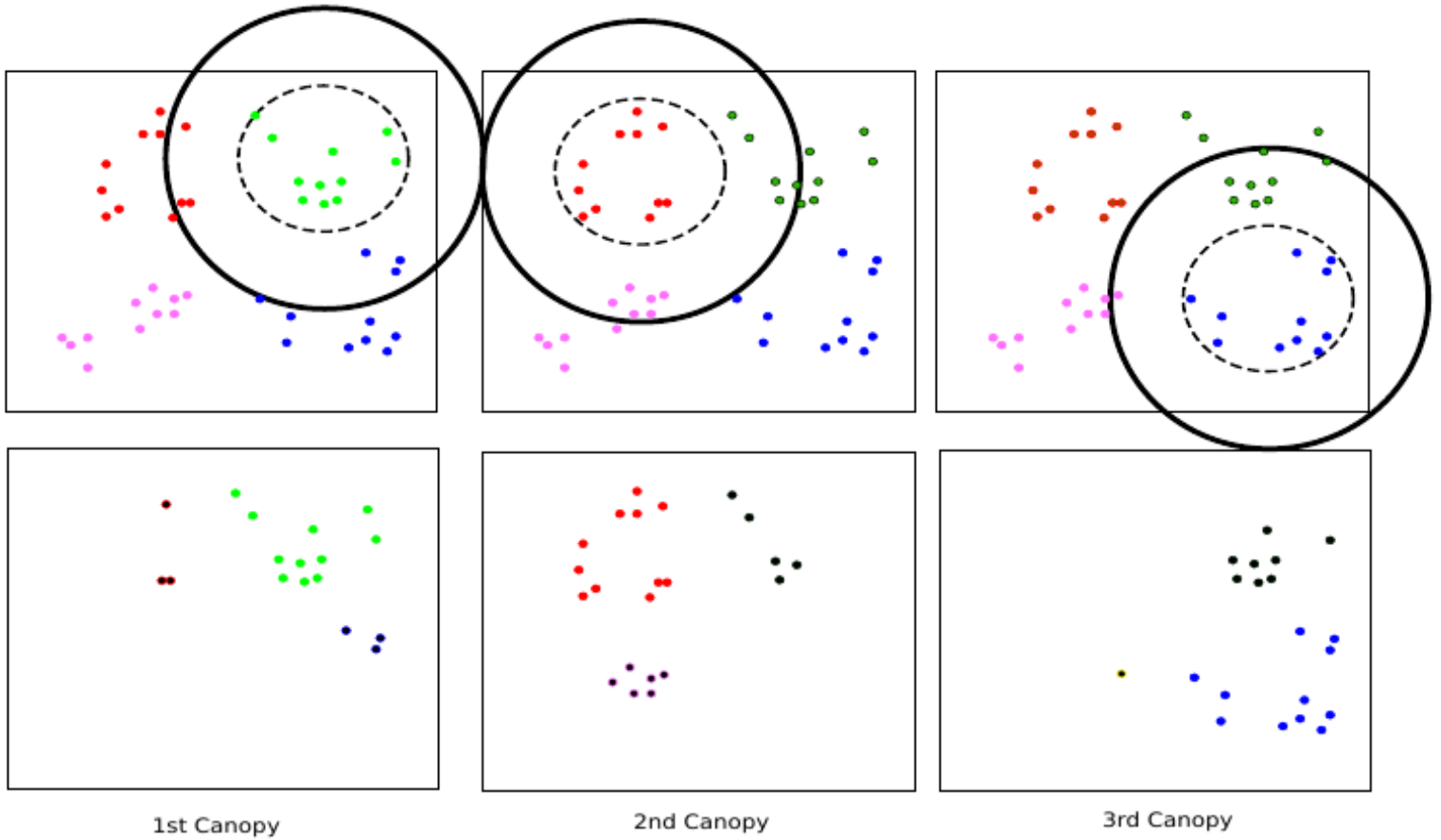
S= Set of Random Points
Choose k random points from S as the initial K-Means Centers, call it set K
while (true)
for all points p in S
  a. for all points k in K
    i. calculate distance between p and k.
  b. Assign p to the k-center which it is closest to.
Re calculate the K-Mean Centers, the new K-Mean centers are an average of all the points assigned to a particular K-Mean Center.
If the calculated error is < allowed error break;

Total runtime is **O(kni) with k: nr. of k-centers, n: nr. of points, i: nr. of iterations to converge**

K-Means with Canopy-Clustering:

S= Set of Random Points
Determine the canopy centers C, and the canopy membership of all points using canopy clustering described above
Choose k random points from S as the initial K-Means Centers, call it set K
while (true)
for all points p in S
  a. for all points k in K
    if p and k share a canopy center
    then calculate distance between p and k.
    else distance is infinite.
  b. Assign p to the k-center which it is closest to.
Re calculate the K-Mean Centers, the new K-Mean centers are an average of all the points assigned to a particular K-Mean Center.
If the calculated error is < allowed error break;

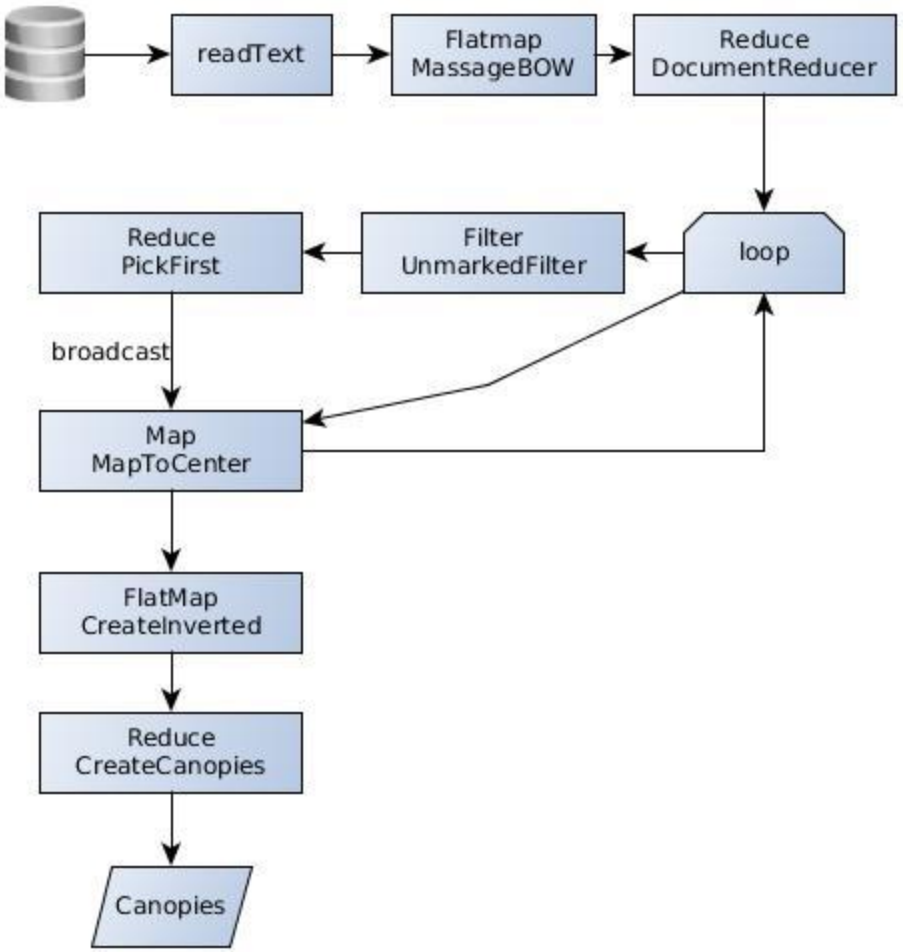
Example



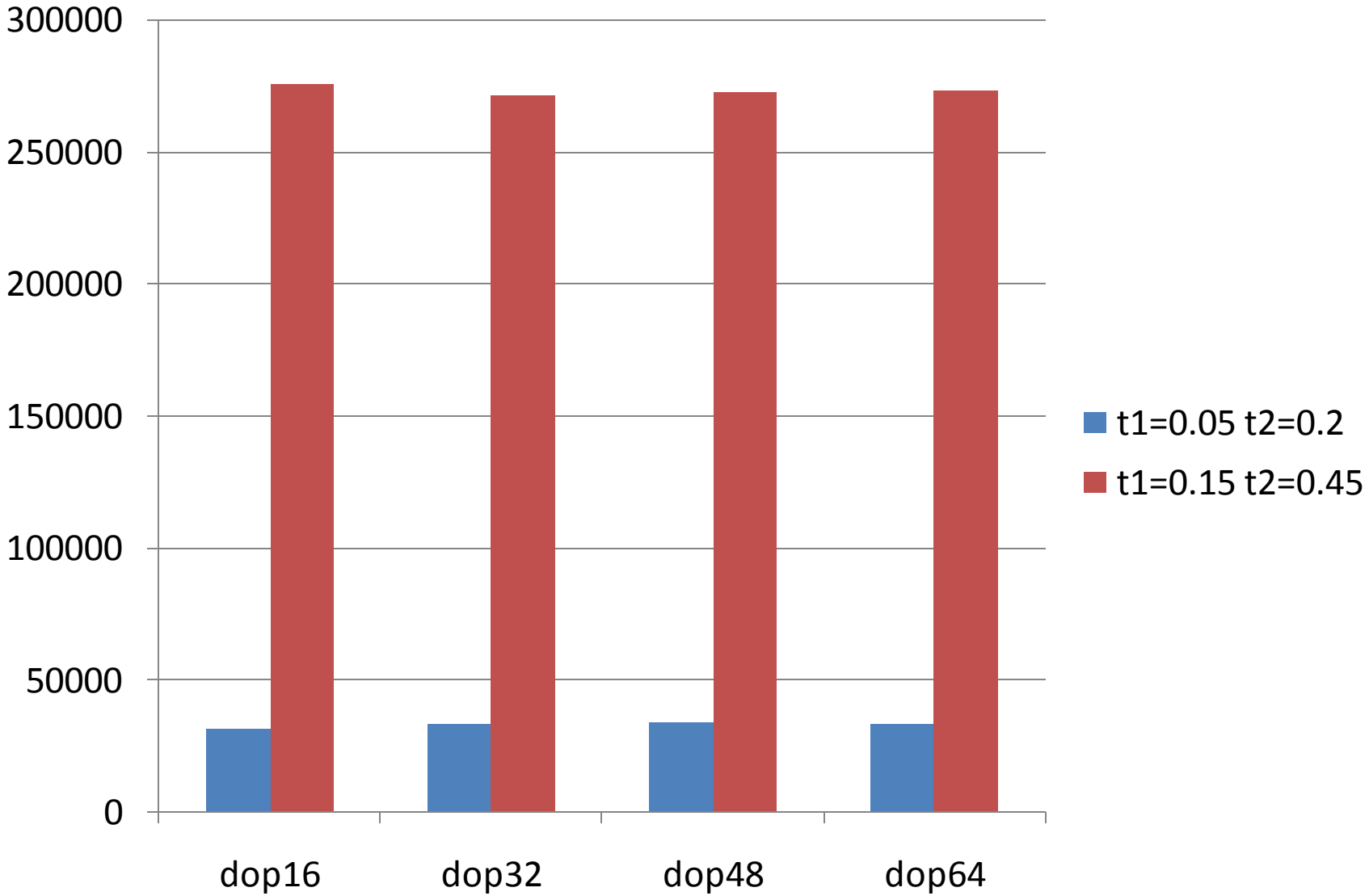
Characteristics

- The number of instances of data that must be compared at second step is reduced (sort of D&C)
- Efficiency and accuracy of canopy depends on chosen threshold
- There is evidence that the runtime of the clustering may improve
- Tradeoff: once a data point has been assigned to canopies, it can never be bound to k-centers in different canopies, even if they would have been better global choices.

Stratosphere



Benchmarks - kos



Sources

- Efficient clustering of high-dimensional data sets with application to reference matching by Andrew McCallum, Kamal Nigam, Lyle H. Ungar
- http://rorlig.wordpress.com/2011/05/13/project-report-cs-292-kmeans-algorithm-with-canopy-clustering/
- http://mahout.apache.org/users/clustering/canopy-clustering.html
- http://www.slideshare.net/mobius.cn/lec4-clustering-3537660
- http://www.lsi.upc.edu/~mmartin/DMClustering.pdf

Thanks