RACHET: A New Algorithm for Clustering Multi-dimensional Distributed Datasets*

Nagiza F. Samatova[a], George Ostrouchov[a], Al Geist[a], and Anatoli V. Melechko[b]

[a]*Computer Science and Mathematics Division*
Oak Ridge National Laboratory **,
P.O. Box 2008, Oak Ridge, TN 37831
{samatovan, ost, geist}@ornl.gov

[b]*Department of Physics and Astronomy,*
University of Tennessee,
Knoxville, TN 37996
melechko@unix.cas.utk.edu

This paper presents a hierarchical clustering method named RACHET (Recursive Agglomeration of Clustering Hierarchies by Encircling Tactic) for analyzing multi-dimensional distributed data. A typical clustering algorithm requires bringing all the data in a centralized warehouse. This results in $O(nd)$ transmission cost, where n is the number of items and d is the number of features. For massive datasets, this is prohibitively expensive. In contrast, RACHET runs with at most $O(n)$ time, space, and communication costs to build a global hierarchy of comparable clustering quality by merging locally generated clustering hierarchies. RACHET employs the encircling tactic in which the merges at each stage are chosen to minimize the volume of a covering hypersphere. For each cluster centroid, RACHET maintains descriptive statistics of constant complexity to enable these choices. RACHET's framework is applicable to a wide class of centroid-based hierarchical clustering algorithms, such as centroid, medoid, and Ward.

1 Introduction

Clustering of multidimensional data is a critical step in many fields including data mining [FHSU96], statistical data analysis [A73, KR89], pattern recognition and image processing [F90], and business applications [BKKPS96]. *Hierarchical clustering* based on a dissimilarity measure is perhaps the most common form of clustering. It is an iterative process of merging (agglomeration) or splitting (partition) of clusters that creates a tree structure called a *dendrogram* from a set of data points. *Centroid-based hierarchical clustering* algorithms, such as centroid, medoid, or minimum variance [A73], define the dissimilarity metric between two clusters as some function (e.g., Lance-Williams [LW67]) of distances between cluster centers. Euclidean distance is typically used.

We focus on the *distributed* hierarchical clustering problem. We create a hierarchical decomposition of massive data sets that are inherently distributed among various sites connected by a network. For practical reasons, the application to distributed and very massive (both in terms of data points and the number of features, or dimensions, for each point) datasets raises a number of major requirements for any solution to this problem:

1) *Qualitative comparability.* The quality of the hierarchical clustering system produced by the distributed approach should be comparable to the quality of the clustering hierarchy generated from centralized data.
2) *Computational complexity reduction*. Asymptotic time and space complexity of a distributed algorithm should be less than or equal to the asymptotic complexity of the corresponding centralized approach.
3) *Scalability.* The algorithms should be scalable with the number of data points, the number of features, and the number of data stores.
4) *Communication acceptability.* The data transfer/communication overheads should be modest. Doing this with minimal communication of data is a challenge.
5) *Flexibility*. If the solution is based on an existing clustering algorithm, then it should be applicable to a wide class of clustering algorithms.
6) *Visual representation sufficiency.* The summarized description of the resulting global hierarchical cluster structure should be sufficient for its accurate visual representation.

Current clustering approaches do not offer a solution to the distributed hierarchical clustering problem that meets all these requirements. Most clustering approaches [M83, DE84, JMF99] are restricted to the centralized data situation that requires bringing all the data together in a single, centralized warehouse. For large datasets, the transmission cost becomes prohibitive. If centralized, clustering massive centralized data is not feasible in practice using existing algorithms and hardware.

* This work has been supported by the MICS Division of the U.S. Department of Energy
** Oak Ridge National Laboratory is managed by UT-Battelle, LLC U.S. D.O.E. under Contract No. DE-AC05-00OR22725.

Dendrogram build-global-dendrogram(Dendrogram $D[]$) {

For each $\{i, j : 0 \le i < j \le |S|\}$ compute $d_{approx}(\vec{c}_i, \vec{c}_j)$

For each $\{i : 0 \le i \le |S|\}$ compute

$NN(i)$ = **find-best-match**(*D*[*i*], $D[]$)

$DISS(i) = d_{approx}(\vec{c}_i, \vec{c}_{idexOf(NN(i))})$

Initialize $GlobalDendogram$

Repeat $|S| - 1$ times

Determine *i* such that *DISS(i)* is minimized

$Dendogram1 \leftarrow D[i]$

$Dendogram2 \leftarrow NN(i)$

Dendrogram3 = **merge-dendrograms**(*Dendrogram1*, *Dendrogram2*)

Update $GlobalDendogram$, each $DISS(i)$ and $NN(i)$ as necessary

return $GlobalDendogram$ }

Figure 1. An efficient algorithm to build a global dendogram.

Distributed clustering approaches necessarily depend on how the data are distributed. Possible combinations are: *vertical* (features), *horizontal* (data points), and *block* fragmentations. For vertically distributed data sets, Johnson and Kargupta [JK99] proposed the Collective Hierarchical Clustering (CHC) algorithm for generating hierarchical clusters. The CHC runs with a $O(|S|n)$ space and $O(n)$ communication requirement, where *n* is the number of data points and |S| is the number of data sites. Its time complexity is $O(|S|n^2)$, and the implementation is restricted to single link clustering. Parallel based hierarchical clustering approaches [O95, DM99] can be considered as a special case of horizontal data distribution. However, these algorithms are tailored to a specific hardware architecture (e.g., PRAM) or restricted to a certain number of processors. Moreover, there is a major distinction between parallel and horizontally distributed approaches: the data are already distributed so that we do not have the luxury of distributing data for optimal algorithm performance as is often done for parallel computation.

We present a clustering algorithm named RACHET that is especially suitable for very large, high-dimensional, and horizontally distributed datasets. RACHET builds a global hierarchy by merging clustering hierarchies generated locally at each of the distributed data sites. Its time, space, and transmission costs are at most linear $O(n)$ in the size of the dataset. This includes only the complexity of the transmission and agglomeration phases and does not include the complexity of generating local clustering hierarchies. Cluster quality of RACHET can be refined by feature set fragmentation and replication of descriptive statistics for cluster centroids. Finally, RACHET's summarized description of the global clustering hierarchy is sufficient for its accurate visual representation that maximally preserves the proximity between data points.

2 The RACHET Algorithm

We make the following assumptions. First, the data are distributed across several sites where each site has the same set of features but on different items. Homogeneity is assumed not only for the type of features of the problem domain but also for the units of measurements of those features. Next, we use Euclidean distance as the measure of dissimilarity between individual points. Finally, the implementation of RACHET assumes a centroid-based hierarchical clustering algorithm, such as centroid, medoid, or minimum variance (Ward's). Fig. 1 presents the core of RACHET.

2.1 Centroid Descriptive Statistics

Selection and effective description of cluster Descriptive Statistics (DS), or summarized cluster representation, is an important step in merging local clustering hierarchies and in visualization of the global hierarchy. DS have to meet a number of major requirements:

- They should occupy much less space than the naive representation, which maintains all objects in a cluster.
- They should be adequate for efficiently calculating all measurements involved in making clustering decisions such as merging or reconfiguration.
- They should be sufficient to visually represent the global hierarchy.

Let $\vec{c} = (f_{c1}, f_{c2}, ..., f_{cd})$ be a cluster centroid of N_c d-dimensional data points $\{\vec{p}_1, \vec{p}_2, ..., \vec{p}_{N_c}\} \subset R^d$ that is defined as $\vec{c} = \frac{1}{N_c} \sum_{i=1}^{i=N_c} \vec{p}_i$. Let p_{ij} denote the j-th component of the data point $\vec{p}_i$, then the j-th component of $\vec{c}$ is defined as $f_{cj} = \frac{1}{N_c} \sum_{i=1}^{N_c} p_{ij}$, $j = \overline{1,d}$. The *Descriptive Statistics* (DS) of the cluster are defined as a 6-tuple $DS(\vec{c}) = (N_c, NORMSQ_c, R_c, SUM_c, MIN_c, MAX_c)$, where

1) N_c is the number of data points in the cluster.
2) $NORMSQ_c$ is the square norm of the centroid, i.e. $NORMSQ_c := \sum_{j=1}^{j=d} f_{cj}^2$
3) R_c is the radius of the cluster defined as the average Euclidean distance from member points to the centroid, i.e. $R_c := \left[\frac{\sum_{i=1}^{N_c} (\vec{p}_i - \vec{c})^2}{N_c} \right]^{\frac{1}{2}}$
4) SUM_c is the sum of the components of the centroid, i.e. $SUM_c := N_c \sum_{j=1}^{j=d} f_{cj}$
5) MIN_c is the minimum value of the centroid components, i.e. $MIN_c := N_c \min_{1 \le j \le d} f_{cj}$
6) MAX_c is the maximum value of the centroid components, i.e. $MAX_c := N_c \max_{1 \le j \le d} f_{cj}$

2.2 Euclidean Distance Approximation

Given two centroids, $\vec{c}_1$ and $\vec{c}_2$, the squared Euclidean distance between them is defined as:

$$d^2(\vec{c}_1, \vec{c}_2) = \sum_{j=1}^{d} (f_{c_1 j} - f_{c_2 j})^2 \tag{1}$$

To compute the Euclidean distance between centroids from different local datasets would require the transmission of all d centroid components. This approach would require transmission of cluster centroids represented by each node of the dendrogram generated at each of the $|D|$ local datasets. This would result in a transmission cost of $O(nd)$, which can be prohibitively high.

Given the DS of each cluster, we can derive an approximated distance between the two cluster centroids. Equation (1) can be expanded as follows:

$$d^2(\vec{c}_1, \vec{c}_2) = NORMSQ_{c_1} + NORMSQ_{c_2} \quad - 2 \sum_{j=1}^{d} f_{c_1 j} f_{c_2 j} \tag{2}$$

If the cross-product term is ignored, then the distance can be approximated by the sum of square norms of the centroids. This results in a significant error. To reduce this error, we can place a non-zero upper and lower bound on the cross-product term:

$$\frac{1}{N_{c_1} N_{c_2}} MIN_{c_1} SUM_{c_2} \le \sum_{j=1}^{d} f_{c_1 j} f_{c_2 j} \le \frac{1}{N_{c_1} N_{c_2}} MAX_{c_1} SUM_{c_2} \tag{3}$$

$$\frac{1}{N_{c_1} N_{c_2}} MIN_{c_2} SUM_{c_1} \le \sum_{j=1}^{d} f_{c_1 j} f_{c_2 j} \le \frac{1}{N_{c_1} N_{c_2}} MAX_{c_2} SUM_{c_1} \tag{4}$$

Inequalities (3) and (4) hold, if each component of the cluster centroid is positive. Taking the maximum of the lower bounds and the minimum of the upper bounds in (3) and (4) leads to the following bounds on the Euclidean distance:

$$d^2_{lower}(\vec{c}_1,\vec{c}_2) = \max\{0, NORMSQ_{c_1} + NORMSQ_{c_2} - 2\frac{1}{N_{c_1}N_{c_2}}\min\{MAX_{c_1}SUM_{c_2}, MAX_{c_2}SUM_{c_1}\}\}$$

$$d^2_{upper}(\vec{c}_1,\vec{c}_2) = NORMSQ_{c_1} + NORMSQ_{c_2} - 2\frac{1}{N_{c_1}N_{c_2}}\max\{MIN_{c_1}SUM_{c_2}, MIN_{c_2}SUM_{c_1}\}$$

Taking the simple mean of the minimum and the maximum square distances gives an approximation of the squared Euclidean distance between two centroids

$$d^2_{approx}(\vec{c}_1,\vec{c}_2) = (d^2_{lower} + d^2_{upper})/2 \quad (5)$$

This approximation can be further improved, if the d components of a cluster centroid are partitioned into the k fragments, and $MIN_{c(k)}$, $MAX_{c(k)}$, and $SUM_{c(k)}$ are maintained for each of the k fragments. In this case, the parameter k will be called the *fragmentation parameter*.

2.3 Merging Two Dendrograms

Given data sets S_1 and S_2 and their dendrograms D_1 and D_2, generated by a hierarchical clustering algorithm applied locally to each data set, Fig. 2 illustrates four different cases (out of six possible) of merging the two dendrograms (Fig. 2.a) into dendrogram D_{new}. Each cluster is represented by a covering hypersphere $(\vec{c}, R_c)$ defined by its centroid $\vec{c}$ and radius R_c. In what follows, the terms "cluster" and its "hypersphere" will be used interchangeably.

Case 1 (Fig. 2.b): This case is designed to merge two well separated datasets. Two clusters, $\vec{c}_1$ and $\vec{c}_2$, are well separated if their hyperspheres do not intersect. That is, $d(\vec{c}_1,\vec{c}_2) \geq R_{c_1} + R_{c_2}$. In this case, a new parent node, D_{new}, is created and dendrograms D_1 and D_2 become the children of the new node. The descriptive statistics of the new cluster are updated. The new cluster centroid is computed as the center of gravity of mass $N_{c_1} + N_{c_2}$ with the radius updated based on the Huyghen's equation [LMW84]. Due to the lack of space, the details on how to update descriptive statistics are omitted.

Case 2: Here the data points of the first cluster are contained in the hypersphere with center $\vec{c}_2$ and radius R_{c_2}, i.e. $d(\vec{c}_1,\vec{c}_2) < R_{c_2}$. This case is further subdivided into two subcases:

Case 2.a (Fig. 2.c): The first cluster $(\vec{c}_1, R_{c_1})$ is well separated from any other child cluster $(\vec{c}_{2j}, R_{c_{2j}})$ of the second cluster $(\vec{c}_2, R_{c_2})$, $j = 1,2,\ldots$. In this case, dendrogram D_1 becomes a new child of dendrogram D_2. The descriptive statistics of $\vec{c}_2$ are updated similarly to Case 1.

Case 2.b (Fig. 2.d): The first cluster $(\vec{c}_1, R_{c_1})$ overlaps with one or more child clusters $(\vec{c}_{2j}, R_{c_{2j}})$ of the second cluster $(\vec{c}_2, R_{c_2})$, $j = 1,2,\ldots$. Here the child cluster that matches best with the dendrogram D_1 is selected to be merged with this dendrogram using a recursive call to the merge_dendrograms() process. There are a number of possible choices for defining a "best match". One choice for the best match is the cluster that has the largest intersection area with the candidate cluster. The new node that is returned by the merge_dendrograms() process replaces the selected child in the dendrogram D_2. If the new node D_{new} has more than two children, then its descriptive statistics are obtained by repeatedly updating two children at a time.

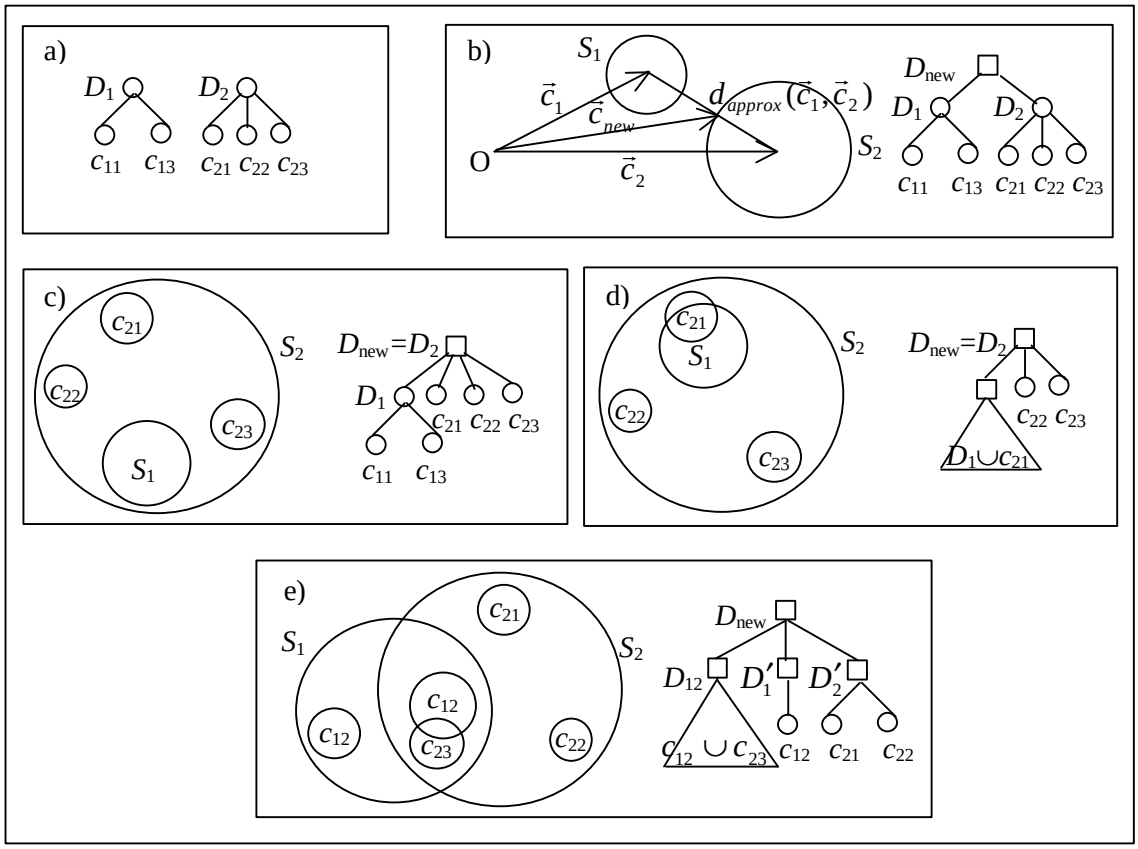

Figure 2. Illustration of the four cases to merge two dendograms. (a) Two dendograms D_1 and D_2. (b) Merging two well separated clusters (Case 1). (c) Making cluster S_1 a subcluster of cluster S_2 provided a proper containment of cluster S_1 in cluster S_2 (Case 2.a). (d) Merging cluster S_1 with the best matched subcluster of clster S_2 provided a proper containment of cluster S_1 in cluster S_2 (Case 2.b). (e) Merging two overlapping clusters (Case 4)

Case 3: This case addresses the situation when data points of the second cluster are contained in the hypersphere with the center $\vec{c}_1$ and the radius R_{c_1}, i.e. $d(\vec{c}_1,\vec{c}_2) < R_{c_1}$. This case is similar to case 2.

Case 4 (Fig. 2.e): This last case is designed to merge partially overlapped clusters, i.e. ($d(\vec{c}_1,\vec{c}_2) < R_{c_1} + R_{c_2}$) and ($d(\vec{c}_1,\vec{c}_2) > R_{c_1}$ or $d(\vec{c}_1,\vec{c}_2) > R_{c_2}$). This case tries to improve the quality of the clustering by reconfiguring the children of both dendrograms D_1 and D_2. We omit the details.

3 Summary and Future Work

This paper has presented RACHET, a hierarchical clustering method for very large, high-dimensional, and horizontally distributed data sets. Most hierarchical clustering algorithms suffer from severe drawbacks when applied to very massive and distributed data sets: 1) they require prohibitively high communication cost to centralize the data to a single site and 2) they do not scale up with the data base size and with the dimensionality of data sets. RACHET makes the scalability problem more tractable. This is achieved by generating local clustering hierarchies on smaller data subsets and using condensed cluster summaries for the consecutive agglomeration of these hierarchies while maintaining the clustering quality. Moreover, RACHET has significantly lower (linear) communication costs than traditional centralized approaches.

In the near future, we plan to concentrate on:

- Error-bounded approximation of Euclidean distance.
- Refinement of RACHET to handle non-spherical shapes for cluster representation, i.e. non-normal and mixed forms to approximate the distribution of data points in the cluster.

- Study of the sensitivity of RACHET's performance to various characteristics of the data. The characteristics include various partitions of data points across distributed sites, clusters of different shapes, sizes, and densities, the number of data sites, different sizes and dimensions of data, and so on.
- Extension of RACHET to handle non-centroid-based hierarchical algorithms as well as non-Euclidean dissimilarity measures.
- Application of RACHET to very large real and synthetic data sets.

References

[A73] Anderberg, M.R., 1973, *Cluster analysis and applications* (Academic Press).

[BKKPS96] Brachman R., Khabaza T., Kloesgen W., Piatetsky-Shapiro G., and Simoudis E., 1996, Industrial Applications of Data Mining and Knowledge Discovery. *Communications of ACM*, **39**.

[DE84] Day W. H. E. and Edelsbrunner H., 1984, Efficient algorithms for agglomerative hierarchical clustering methods. *Journal of Classification*, **1**, 7-24.

[DM99] Dhillon I. and Modha D., 1999, A data clustering algorithm on distributed memory multiprocessors. In *Workshop on Large-Scale Parallel KDD Systems*.

[FHSU96] Fayyad U., Haussler D., Stolorz P., and Uthurusamy R. (Eds.), 1996, *Advances in Knowledge Discovery and Data Mining* (MIT Press).

[F90] Fukunaga K., 1990, *Introduction to Statistical Pattern Recognition* (Academic Press).

[JMF99] Jain A.K., Murty M.N., and Flynn P.J., 1999, Data Clustering: A Review. *ACM Computing Surveys*, **31**, 264-323.

[JK99] Johnson E. and Kargupta H., 1999, Hierarchical clustering from distributed, heterogeneous data. *Lecture Notes in Computer Science*, **1759** (Springer-Verlag).

[KR89] Kaufman L. and Rousseeuw P., 1989, *Finding Groups in Data* (John Wiley and Sons).

[LW67] Lance G.N. and Williams W.T., 1967, A general theory of classificatory sorting strategies. 1: Hierarchical systems. *Computer Journal*, **9**, 373-380.

[LMW84] Lebart L, Morineau A., and Warwick K., 1984, *Multivariate descriptive statistical analysis: Correspondence analysis and related techniques for large matrices* (John Wiley & Sons).

[M83] Murtagh F., 1983, A survey of recent advances in hierarchical clustering algorithms. *Computer Journal*, **26**, 354-359.

[O95] Olson C., 1995, Parallel algorithms for hierarchical clustering. *Parallel Computing*, **8**, 1313-1325.