Hierarchical Clustering Techniques From Mixed Data Sequences

Dinesh Bhardwaj and Dr. Sonawane Vijay Ramnath

Department of Computer Science & Engineering, Dr. A. P. J. Abdul Kalam University, Indore (M.P.) – 452010

Corresponding Author Email: dkbh28@gmail.com

Article Info
Page Number: 12644-12651
Publication Issue:
Vol. 71 No. 4 (2022)

Article History
Article Received: 15 October 2022
Revised: 25 November 2022
Accepted: 15 December 2022

Abstract

Through the efficient organization of massive volumes of data into a small number of relevant clusters, high-quality document clustering algorithms play a crucial role in facilitating straightforward navigation and browsing procedures. Data stream hierarchical clustering approaches are covered, as with comparisons of algorithmic performance. Furthermore, this study explains and compares many data clustering algorithms. The standard datasets are used as input, and the appropriate hierarchical clustering technique is then used to them. The output should be clustered data that is well-versed and appropriately ordered. Microbial community analysis relies heavily on taxonomy-free methods of investigation. Many subsequent studies rely on identifying operational taxonomic units, and hierarchical clustering is one of the most popular methods for doing so. Because of their quadratic space and computing difficulties, most known methods are limited in their applicability to situations of moderate size or less. To solve the space and computational bottlenecks of existing solutions, we offer a novel online learning-based technique.

Keywords: Clustering algorithms,data streams clustering, hierarchical clustering, parameter selection

Introduction

Many fields are keenly interested in hierarchical clustering solutions, which take the shape of trees known as dendrograms. Data may be seen at many levels of abstraction with the use of hierarchical trees. Flat partitions of varying granularity may be derived from clustering algorithms during data analysis, making them perfect for interactive exploration and visualization. The hierarchical structure is definitely a natural constraint on the underlying application area (e.g., biological taxonomies, phylogenetic trees, etc.), and there are numerous cases in which clusters include sub clusters. The majority of successful hierarchical clustering solutions have been produced by using agglomerative algorithms, in which items are first placed in their own cluster before further merging pairs of clusters to build the tree structure. However, hierarchical clustering solutions may also be obtained using partitional algorithms by a series of recurrent bisections.

As a result of their low processing needs, many academics have come to agree that partitional clustering methods are ideal for the task of clustering massive document collections. However, it is widely held that partitioned algorithms perform worse than their agglomerative equivalents when it comes to clustering quality. The evidence for this claim comes from a few research and several trials on low-dimensional datasets, which showed that agglomerative methods often outperformed partitioned K-means based methods. This is why prior studies of hierarchical document clustering techniques concentrated on agglomerative approaches while ignoring partitioned techniques. Furthermore, much of the existing research has assessed the efficacy of different clustering algorithms by gauging the effectiveness of the resultant clustering solutions in enhancing retrieval. Evaluations of the generated hierarchical trees' consistency with preexisting class information are scarce and rely on a small number of datasets.

We compare six recently studied criterion functions for partitional clustering, all of which have been shown to yield high-quality solutions, and we study three classic merging criteria for agglomerative clustering (i.e., single-link, complete-link, and group average (UPGMA)) as well as a new set of merging criteria introduced in this paper that are derived from the six criterion functions. We found that most partitional approaches provide hierarchical clustering solutions that are consistently and noticeably superior than those yielded by the different agglomerative algorithms. Due to their superior performance in terms of cluster quality and cheap processing needs, our results show that partitional clustering techniques are excellent for generating hierarchical solutions of large document datasets. Second, we develop partitional clustering methods to produce intermediate clusters that we use to restrict the choice space for agglomeration techniques.

Literature Review

Jerry W. Sangma (2022) There have been several efforts in data stream clustering over the last decade, although most of these studies are classified as clustering by example approaches. Clustering of different data streams, as opposed to clustering data instances within a single data stream, is necessary for a variety of applications, which necessitates a "clustering by variable method." Additionally, a few studies have been published for multi-stream clustering, however they only operate with numerical data. Therefore, this knowledge gap has prompted ongoing studies. In this paper, we offer a hierarchical clustering method for aggregating data from various streams in cases when the data are notional. Splitting and merging clusters in a hierarchical structure is carried out to deal with idea shifts in the data streams. Based on the entropy measure, which represents the degree of dissimilarity inside the cluster, a decision is made to either separate or combine the groups. Dunn Index, Modified Hubert statistic, Cophenetic Correlation Coefficient, and Purity measurements were used to evaluate the suggested method's efficacy on both a simulated and real-world dataset. For data streams that represent developing concepts, the suggested method outperforms the Agglomerative Nesting clustering method. In addition, we now have a visual representation with which to examine and comprehend the impact of idea development on clustering structure and average entropy.

PranavShetty (2021) Examining data and extracting useful insights is essential in the modern era. Clustering is an analytical method that involves categorizing data into sets of records that share similar characteristics. Clusters are the building blocks of every group, and its members share similarities with others in their cluster yet stand out from the crowd in important ways. Our goal in this research is to examine and contrast two distinct hierarchical clustering approaches. Clustering may be done in a variety of ways, the most common being partition and hierarchical clustering. One of the methods covered is the hierarchical clustering technique. Dataset size, data set type, number of clusters produced, consistency, precision, and efficiency are only few of the metrics against which the aforementioned techniques are measured and compared. It is the goal of the cluster analysis method known as hierarchical clustering to establish a tree-like structure in the resulting clusters. Simply put, a hierarchical clustering method is a tree-like organization of multiple, independent, simple (flat) clustering techniques. Recursively dividing the entities either from the top down or the bottom up, these techniques produce clusters. In this paper, we look at and contrast various hierarchical clustering algorithms. The goal of this article is to provide new researchers and novices with a foundational understanding of hierarchical clustering algorithms by describing the many implementations of these algorithms.

Anna Arutyunova (2022) Understanding the genetic properties of a dataset is a common goal of using Hierarchical Clustering. Such a clustering is, in fact, a chain of clusterings that begins with the trivial clustering, in which each data point forms its own cluster, and then merges two existing clusters, one after the other, until all points are in the same cluster. If the costs of each k-clustering in the hierarchy are no more than times the costs of an optimal k-clustering, then the hierarchical clustering achieves an approximation factor of. We investigate the maximum (discrete) radii and diameters of clusters as cost functions for the k-center problem. In most cases, the best clustering options do not arrange themselves in a hierarchy, making it impossible to achieve a precision of 1 in approximation. The price of hierarchy is defined as the smallest achievable approximation factor for any given instance. We reduce the cost of hierarchy up to 3+22-5.83, for the k-diameter problem. We also prove a price of hierarchy of precisely 4 and 3+22- for k-center and k-diameter, respectively, vastly improving previous lower bounds for these quantities.

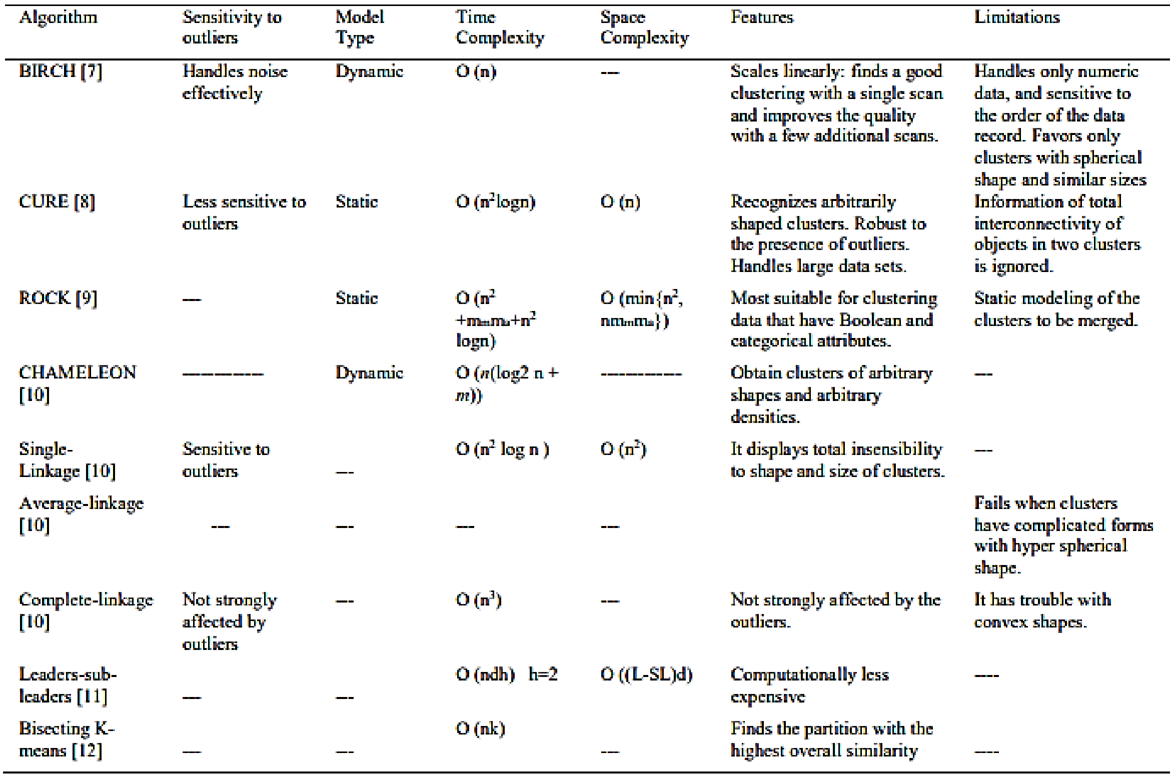
Zixiang Pan (2022) Research into complicated bio-tissues may be improved with the use of single-cell sequencing tools, which allow for a more nuanced analysis of cellular composition. However, subsequent efforts have focused on identifying even finer subtypes within these established cell types. We introduce MeHi-SCC, a methodology that incorporates meta-learning and incorporates data from several scRNA-seq datasets to aid in the graph-based hierarchical sub-clustering process. MeHi-SCC identified cell subtypes in two large-scale cell atlases with more accuracy than previously available scRNA clustering approaches.

Ana Radovanović (2020) As technology has improved, it has been feasible to automatically capture and store vast amounts of data, creating a demand for more refined methods of data analysis. Information may be gleaned from datasets of varying shapes and sizes using unsupervised data clustering techniques. Here, we use two time series datasets, one from the power grid and the other from the neuroscience field, to analyze the efficacy of the prevalent agglomerative hierarchical clustering approach. Clustering's primary stages are laid out here for your perusal. The results demonstrate that the primary properties of the clustering data and the algorithm's settings significantly impact the algorithm's performance.

Methodology

"For a given set of data points, partition them into one or more groups of similar objects," is how the clustering problem is formally stated. Distance measures or objective functions are often used to determine how similar two items are to one another. This section explores a few of the many available clustering approaches. If k = n, then the data is partitioned into k partitions, each of which represents a cluster. Distance between objects is the basis for many different partitioning techniques. It organizes the information into k groups where all items are represented and where each item belongs to exactly one group. It all starts with an initial partitioning, which is generated using a partitioning technique. Then, it employs an iterative relocation technique to see if shifting objects around helps the partitioning. The objects within the same cluster should be 'close' or related to each other, while the objects within different clusters should be 'far apart.'

Table 1: Comparison of hierarchical clustering algorithms



Algorithm	Sensitivity to outliers	Model Type	Time Complexity	Space Complexity	Features	Limitations
BIRCH [7]	Handles noise effectively	Dynamic	O (n)	---	Scales linearly: finds a good clustering with a single scan and improves the quality with a few additional scans.	Handles only numeric data, and sensitive to the order of the data record. Favors only clusters with spherical shape and similar sizes
CURE [8]	Less sensitive to outliers	Static	O (n^2logn)	O (n)	Recognizes arbitrarily shaped clusters. Robust to the presence of outliers. Handles large data sets.	Information of total interconnectivity of objects in two clusters is ignored.
ROCK [9]	—	Static	O ($n^2 + m_m m_a + n^2$ logn)	O (min{n^2, $n m_m m_a$})	Most suitable for clustering data that have Boolean and categorical attributes.	Static modeling of the clusters to be merged.
CHAMELEON [10]	————	Dynamic	O (n(log2 n + m))	----------	Obtain clusters of arbitrary shapes and arbitrary densities.	---
Single-Linkage [10]	Sensitive to outliers	---	O (n^2 log n)	O (n^2)	It displays total insensibility to shape and size of clusters.	---
Average-linkage [10]	---	---	---	---		Fails when clusters have complicated forms with hyper spherical shape.
Complete-linkage [10]	Not strongly affected by outliers	---	O (n^3)	---	Not strongly affected by the outliers.	It has trouble with convex shapes.
Leaders-sub-leaders [11]	—	---	O (ndh) h=2	O ((L-SL)d)	Computationally less expensive	----
Bisecting K-means [12]	—	---	O (nk)	---	Finds the partition with the highest overall similarity	----

A group of data items may be hierarchically decomposed using a hierarchical clustering technique. Specifically, it can be broken down into two subtypes: agglomerative hierarchical clustering and divisive hierarchical clustering. An agglomerative method, also called a bottom-up method, starts with the assumption that each object belongs to a distinct cluster. Next, it merges clusters that are relatively close together, and so on, until all objects belong to a single cluster. The following are the stages of an agglomerative hierarchical clustering.

1. Initiate with a single idea

2. Apply a recursive addition to two or more appropriate clusters.

3. If k clusters are created, the process ends there.

If you're using a top-down, or polarizing, approach, you're lumping everything together from the outset. Then, at each iteration, it breaks into smaller clusters until all objects are in a single cluster or a termination condition is fulfilled. The steps below outline what is involved in discrepant hierarchical clustering.

1. Begin with a hefty clump.

2.is a recursive process that creates subclusters.

3. Once k clusters have been formed, the procedure ends.

Following is a list of the generic procedures required by any hierarchical clustering technique.

1. If you have N items, divide them up into N clusters, each of which will contain a single item. Give each group of objects the same amount of space between clusters as there is between the objects themselves.

2. Locate the two clusters that are geographically nearest to one another, then combine them into a single Cluster.

3. Determine how far apart the newly formed cluster is from the old clusters.

4. Iterate steps 2 and 3 until all objects fit into a single cluster of size N..

Many academics and practitioners rely on hierarchical clustering because it allows them to readily see the results using a dendrogram without having to define a desired number of groups (graphical representation). No undoing the merger or break of a hierarchical cluster. Since there is no need to account for a combinatorial explosion of possible decisions, the reduced computing costs brought about by this rigidity are a net positive.

Density-based clustering refers to a subset of clustering techniques that were designed with density in mind. As long as there is an abundance of items or data points in the area, the cluster will continue to expand. An approach like this may be used to eliminate anomalies and find clusters of any size or form. For instance, DBSCAN, OPTICS, and DENCLUE are all instances of such algorithms. The object space is quantized into a fixed number of cells in a grid-based clustering approach. All clustering processes are executed in a grid layout. The key benefit of this method is the short amount of time it takes to analyze data, which depends only on the number of cells in each dimension of quantized space and not on the total number of objects. Two such algorithms are STING and Wave Cluster. To determine the greatest possible match between the data and the hypothesized model, a model-based clustering approach creates hypotheses about each cluster. This method finds clusters by generating a density function that represents the data's actual geographical distribution. The number of clusters is computed mechanically using a well-used statistic. By using outliers in their analysis, they create a more stable clustering process. Methods that use models to group items together include EM, COBWEB, and SOM. In the constraint-based clustering method, constraints are either provided by the user or are determined by the application context, and then the clustering process is carried out. A user's expectation or the intended qualities of the clustering results may be effectively communicated to the clustering process via the usage of constraints. Clustering algorithms may be selected based on the nature of the data and the needs of the application. It can be challenging to categorize a given clustering algorithm as exclusively belonging to one clustering method category due to the fact that some clustering algorithms integrate the ideas of several clustering methods.

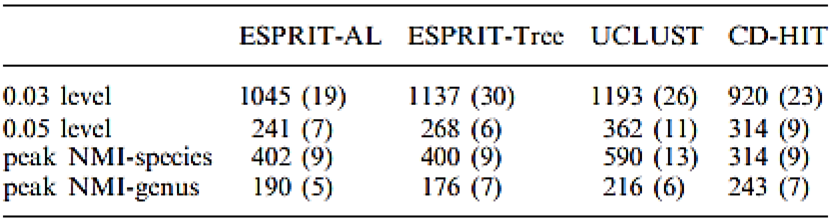
Result & Discussion

The suggested technique depends heavily on the premise that sequence data exists in a pseudo metric space. To back up the aforementioned hypothesis, we ran a simulation research. To begin, we randomly selected 30 K sequences from the gut data set and used ESPRIT with the average linkage function (ESPRIT-AL) to cluster them into groups at distances ranging from 0.01 to 0.10. Then, we picked three clusters at random and used the Needleman-Wunsch method to build three probabilistic sequences (x, y, and z) from the sequences included in those clusters. D(y, z) = d(x, y) + d(x, z)) is the ratio of the pairwise distances of the three sequences. The triangle inequality is true if and only if the ratio is smaller than 1.

We performed this experiment a total of 100 thousand times and found that the inequality was broken just seven times. The results of this experiment seem to support the idea that the triangle inequality is a good approximation for sequence data. The next thing we did was do a benchmark study to see how the four different approaches clustered data. Inspecting the curves, we find that they are all bell-shaped. This is because sequences of the same species are separated into various clusters when the distance threshold is low, while sequences of different species are lumped together when the threshold is high, both of which are certain to provide inferior NMI ratings. We also notice that the four approaches' NMI scores may reach their maximum values in various places owing to variations in the formulas used to calculate the distance between two clusters. This means that it is not possible to do a direct comparison between NMI values obtained at the same distance level. Therefore, we compared the highest possible NMI score for each approach, which, by definition, is the best clustering result that can be achieved.

Again, using the genus identifications as the gold standard, we found the same thing. Estimating the richness of a microbial community is a primary motivation for doing taxonomy-independent analyses. Although 3% and 5% thresholds are routinely employed to determine species and genus-level OTUs in the microbiology literature, they are disputed. The estimated number of species and genera at 0.03 and 0.05 distances, as well as the locations where NMI scores peak, are shown in Table 2. Although all approaches were applied to the identical data sets, we notice that the quantities of OTUs seen at the 0.03 and 0.05 distance levels are much bigger than the ground truths and vary significantly from one another. Several sequencing-error-correction techniques have been developed to address this problem because it was hypothesized that sequencing mistakes are the primary cause of significant overestimation of microbial diversity. Table 2 shows that the estimates of OTUs at the peaks for each approach are consistently more accurate than those at the 0.03 and 0.05 distance levels. This indicates that inaccurate distance levels may be to blame for part of the overestimation. Researchers should exercise caution when interpreting diversity estimates when using the commonly used thresholds of 3% and 5%, respectively, because these thresholds are not appropriate for defining species- and genus-level OTUs. The ESPRIT-Tree and ESPRIT-AL algorithms provided the most reliable estimations of microbial diversity. High throughput pyrosequencing methods create vast amounts of data, presenting significant hurdles to current data analysis techniques.

Table 2: The numbers of OTUs observed at the 0.03 and 0.05 distance levels and at the peak positions for the four methods



	ESPRIT-AL	ESPRIT-Tree	UCLUST	CD-HIT
0.03 level	1045 (19)	1137 (30)	1193 (26)	920 (23)
0.05 level	241 (7)	268 (6)	362 (11)	314 (9)
peak NMI-species	402 (9)	400 (9)	590 (13)	314 (9)
peak NMI-genus	190 (5)	176 (7)	216 (6)	243 (7)

Approximately 377 species and 170 families may be found throughout the world. One standard deviation is represented by the number in parentheses. The most precise estimations of microbial diversity were obtained using ESPRIT-Tree and ESPRIT with the average linkage function (ESPRIT-AL).

The computational complexity of the solution is also a crucial factor to think about. Using a human gut data set containing anything from one thousand to one and a half million sequences, we compared ESPRIT-Tree against CD-HIT and UCLUST to showcase the novel method's scalability feature. Running ESPRIT on 1.1M sequences on a desktop PC is computationally infeasible. There is also a summary of the empirical complexity and associated confidence interval. For its computational efficiency, UCLUST ranks first, followed by ESPRIT-Tree and CD-HIT. Notwithstanding, the computing complexity of all three approaches is O (quasi-linear) (N1.2). Processing 1.1 M reads to construct OTUs at 10 distance levels took ESPRIT-Tree 11 hours (0.01–0.1). Using a cluster of 100 computers, we had previously used ESPRIT on the identical gut data set (12). To complete the study, ESPRIT needed 4 days, making it around 800 times slower than ESPRIT-Tree. Additional tests employing various hypervariable region and almost full-length 16S rRNA sequences yielded the same findings. Results are included in the Supplementary Data because of space constraints.

Conclusion

In this paper, we conducted an experimental evaluation of nine agglomerative algorithms and six partitioned algorithms to find hierarchical clustering solutions for document datasets. By imposing boundaries on the agglomeration process via clusters obtained via partitioned algorithms, we also introduced a new category of agglomerative algorithms. In this work, the efficacy of hierarchical data stream clustering is measured and analyzed. Precision, recall, purity, G-precision, G-recall, and other similar measures are used to evaluate performance. To get more accurate OTU counts from real data (where chimeras are common), it is possible to check the tree's output for chimeras. Despite the article's primary focus on 16S rRNA-based studies, the new algorithm can be applied to other large-scale sequence-based studies that necessitate large-scale clustering analyses.

REFERENCE

1. Sangma, J.W., Sarkar, M., Pal, V. et al. Hierarchical clustering for multiple nominal data streams with evolving behaviour. Complex Intell. Syst. 8, 1737–1761 (2022).
2. Anna Arutyunova (2022) “The Price of Hierarchical Clustering∗” arXiv:2205.01417v1 [cs.DS] 3 May 2022
3. Zixiang Pan (2022) “A Meta-learning based Graph-Hierarchical Clustering Method for Single Cell RNA-Seq Data” bioRxiv preprint doi: https://doi.org/10.1101/2022.09.06.506784
4. Shetty, Pranav& Singh, Suraj. (2021). Hierarchical Clustering: A Survey. International Journal of Applied Research. 7. 178-181. 10.22271/allresearch.2021.v7.i4c.8484.
5. Radovanovic, A., Li, J., Milanovic, J. V., Milosavljevic, N., &Storchi, R. (2020). Application of Agglomerative Hierarchical Clustering for Clustering of Time Series Data. In Proceedings of 2020 IEEE PES Innovative Smart Grid Technologies Europe, ISGT-Europe 2020 (pp. 640-644). [9248759] (IEEE PES Innovative Smart Grid Technologies Conference Europe; Vol. 2020-October).
6. Bones CC, Romani LA, de Sousa EP (2016) Improving multivariate data streams clustering. In: EmbrapaInformáticaAgropecuária-Artigoemanais de congresso (ALICE), Procedia Computer Science, pp 461–471
7. S. Aghabozorgi, A. SeyedShirkhorshidi, and T. Ying Wah, “Time series clustering – A decade review,” Inform. Syst., vol. 53, pp. 16-38, October – November 2015.
8. C. C. Aggarwal, C. K. Reddy, Data Clustering: Algorithms and Applications. Florida, USA: CRC Press, 2013.
9. B. Mohamad, D. Usman, “Standardization and its effects on k-means clustering algorithm,” Res. J. Appl. Sci. Eng. Tech., vol. 6, no. 17, pp. 3299-3303, September 2013.
10. Ackermann MR, Märtens M, Raupach C, Swierkot K, Lammersen C, Sohler C (2012) Streamkm++ a clustering algorithm for data streams. J ExpAlgorithmics (JEA) 17:1–2
11. J. Han, M. Kamber, J. Pei, Data Mining Concepts and Techniques, 3rd ed. Waltam, USA: Elsevier, 2012.
12. Balzanella A, Lechevallier Y, Verde R (2011) Clustering multiple data streams. In: New perspectives in statistical modeling and data analysis, Springer, pp 247–254
13. S. L. Everitt, M. Leese, D. Stahl, Cluster analysis, 5ht ed. London, UK: John Wiley & Sons, Ltd., 2011.
14. Y. S. Jeong, M. K. Jeong, O. A. Omitaomu, “Weighted dynamic time warping for time series classification,” Pattern Recognit., vol. 44, no. 9, pp. 2231-2240, September 2011.