We have taken the topic of Scalable Hierarchical clustering which is one of the interesting topic when it comes to grouping or clustering of the data. In this paper SCC algorithm has been designed and is been tested on very large datasets.

INTRODUCTION

- Clustering is a technique that is widely used in today's world to produce similarities by analyzing and visualizing large datasets.
- There are mainly two types of clustering techniques, Hierarchical (Bottom-up approach) and Flat Clustering (Top-down approach)
- In class, we were introduced to the most common Flat clustering technique, K-means clustering, a centroid-based algorithm. Using the fundamentals of the clustering concepts, in this paper we present a novel scalable and hierarchical agglomerative clustering algorithm that is not affected by the size of the dataset and does not compromise the quality of clustering or the accuracy.

In class we learnt about a Flat clustering technique which is K-means clustering. Based on the understanding about clustering we tried to learn about a new clustering technique which is a scaled version of Hierarchical clustering.
Hierarchical clustering is a bottom-up approach. It has several advantages; we can easily see possible linkages between the clusters. We don't need to preset the number of clusters beforehand as we did in the K-means clustering. In hierarchical clustering we will be able to visualize the data in the form of dendrogram which helps in understanding of data much better. Hierarchical clustering has its disadvantages as well, it is extremely computationally expensive when dealing with large datasets as compared to k-means clustering. In this paper we are studying about a scaled version of the Hierarchical clustering where the authors of our selected paper are trying to improvise the hierarchical clustering such that, it can perform effectively

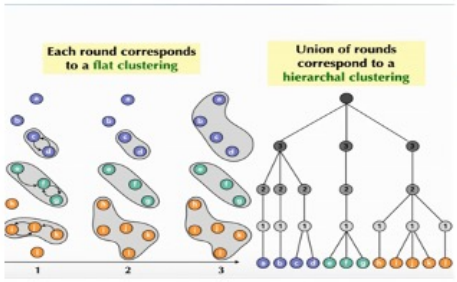
even with large datasets.

Sub-cluster Component Algorithm (SCC)

- SCC is a clustering algorithm that produces a compact, non-binary tree structure where leaves are the data points, and the internal nodes are the clusters formed after each round.
- The algorithm starts as all the data points are its own clusters and there is an input received which is monotonically increasing sequence of thresholds.
- Every node connects to its nearest neighbor and then we prune those edges based on the threshold value. If the distance between any two nodes is greater than the threshold, we remove such edges.

In this slide we are explaining about the SCC algorithm. It is an algorithm which produces a non – binary tree. In this algorithm as a first step, we will treat all data points as its own cluster, and we will receive input thresholds which will be in an increasing sequence. In each step we will be calculating the distance using linkage function and if the value of the linkage function is less than or equal to threshold then we will add data points into a cluster and if it is exceeding the threshold then it will be discarded. Similarly, this process will repeat until we reach a point where we are unable to make any additional changes, thus terminating the algorithm

SCC (continued)



The Sub-Clustering Component Algorithm. We illustrate SCC on a small dataset. The formation of subclusters is shown with black arrows for pairs of points. The direction indicates the nearest neighbor relation- ship. Red edges indicate the nearest neighbor relationships that are above the distance thresholds. The grey circles indicate the sub-cluster components created in that round. Best viewed in color.

- We pop the lowest thresholds and merge the clusters from the previous round such that distance between the two clusters is less than or equal to the popped threshold value.
- For the remaining graph we merge the connected components and then advance on to the next round. We continue as such until we find that there is no change in consecutive rounds and then we advance our threshold to the next large value. We continue in the above manner until we have no more clusters remaining to merge.

The above image represents the linkage and the clustering of Hierarchical clustering. Like HAC, SCC inherits the best-first manner, it puts together points in a cluster in rounds. The sequence of rounds begins with the decisions that are “easy to make” (e.g., points that are clearly in the same cluster) and prolongs the later, more difficult decisions until these confident decisions have been well established. We can see that data points are clustered based on the linkage between their closest neighbors. Like, this the process will be repeated until we reach the last stage of algorithm.

RIT

Comparison between SCC and Hierarchical Agglomerative Clustering (HAC)

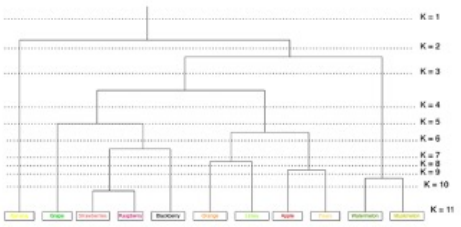


Figure 2: HAC representation of an arbitrary fruits' dataset.

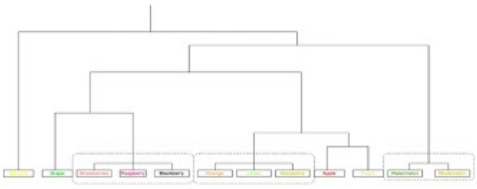
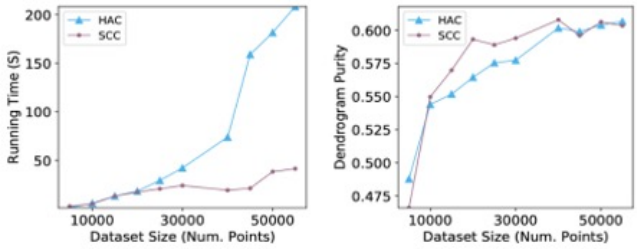


Figure 3: SCC representation of a dataset similar to the fruit dataset used to visualize HAC. The Dotted Rectangles indicated that the datapoints are in a subcluster.

- Let's take a dataset of fruits and cluster them using their types/family like Berries, Tropical, Citrus, Melons, etc. The fruits belonging to the same family should belong in the same cluster.
- HAC creates binary tree making decisions as it goes from bottom to top, while this algorithm is not particularly suited for large datasets as due to its sequential nature and large internal node count.
- Binary trees only accommodate up to two children and this increases the space complexity of the algorithm as shown in Figure 2. Therefore, HAC has limited scalability to larger and more complex datasets.
- Whereas on the other hand, SCC uses non-binary tree structure and is much more scalable compared to HAC.
- As seen in the figure 3 using the same dataset only with some extra data but the same clustering criteria and find that the tree has a lot less internal nodes and is much compact.

Difference between the typical HAC vs the scalable HAC.



- HAC has quadratic time complexity and grows at a faster rate as the size of the dataset increases, whereas, on the other hand, SCC follows a close to linear characteristics. But still, the dendrogram purity of the SCC trees is pretty much like those produced by HAC. The only difference between the tree structure is HAC produces binary trees and SCC produces non-binary trees.

Purity in dendrogram is considered as a measure of range to which clusters contain a single class. In other words, Dendrogram purity can be given as the mean, over all pair of points from the same ground cluster, of the purity of the least common ancestor of the pair. Most clustering algorithms, try to maximize dendrogram purity as it the main goal of clustering is to construct a tree that puts similar points closer together in the tree. Among some of the taken datasets the SCC algorithm performs best by attaining maximum dendrogram purity for all but one considered dataset. For this algorithm, we can consider either linear or geometric progression for threshold. After noticing the algorithm output or performance an observation is made where SCC performs a bit finer in terms of geometric progression compared to linear.

WORKING EXAMPLE

- SCC algorithm prefers to work on large datasets, so let's take the IMDB dataset which has about 8 million entries in their movies dataset which has movies of different genres like Comedy, Action, Sci-fi, etc.
- Using SCC, we extract and visualize the dataset by dividing each movie into sub-cluster of genres. SCC will generate cluster which are accurate, clear and coherent as shown in Figure.
- We represent the hierarchy using rectangular boxes. The root with header is represented by the outer rectangular box (solid line). The second level of the hierarchy, with header, is shown in dashed (– –) rectangle withing the outer box.
- Finally, the third level from the root is shown as the inner most dotted (...) rectangle. Plain text within each of the dotted rectangle are the families that belong to that cluster. We can add as a note that there will be many sub-clusters depending on the dataset and the selection criteria.

Conclusion

- In this paper, a new algorithm Sub-Cluster Component algorithm (SCC) for scalable clustering was introduced which uses best-first, round based, agglomerative method.
- In each round, the distance threshold is increasing which dictates which points can be merged to make a subcluster. Due to merging, the tree structure of SCC is compact and consume less space because the trees are non-binary leading to less levels compared to the tree for HAC.
- However, there exists some threshold value for which no internal merging occurs and the trees generated by SCC is like that of HAC.
- We later compared, the performance of HAC and SCC in terms of their running times on large dataset size and dendrogram purity. The performance of SCC was found to be increasing in at a slower rate to HAC, whereas the dendrogram purity is similar.

We would like to thank Professor Carlos for giving this group activity where we learnt about a new algorithm, and we were able to understand the paper easily based on some related topic knowledge from the class.