Survey on Cooperative Clustering Models

Diya Davis
Post-Graduate Student
Department of Computer Science and Engineering, Karunya University
India

Bright Gee Varghese
Assistant professor
Department of Computer Science and Engineering, Karunya University
India

Abstract

Clustering is the process of finding similar groups of data entities. The important application of clustering is to modularize a software system by grouping together related software entities, thereby providing a high-level view of the system. A number of clustering algorithms and measures have been proposed and applied to obtain a better software modularization. But a single clustering method cannot adequately handle all sorts of cluster structures and properties such as overlapping, shape, size and density. Cooperative clustering (CC) model involves cooperation among multiple clustering techniques for the goal of increasing the homogeneity of objects within the clusters. The cooperative clustering models are capable of showing significant improvement over individual clustering algorithms for software modularization.

1. Introduction

Each clustering algorithm works on its domain space with no optimum solution for all datasets of different properties, sizes, structures, and distributions. The cluster structure produced by a clustering method is sometimes an artifact of the method itself that is actually imposed on the data rather than discovered about its true structure. A single clustering method cannot effectively deal with all kinds of cluster structures and configurations. Therefore clustering methods can be combined to improve the clustering results. Combining different clustering methods invoke multiple clustering algorithms in the clustering process to achieve global benefit. All the methods cooperate together to attain better overall clustering quality.

Multiple clustering algorithms can be combined in the clustering process to benefit from each other to achieve global benefit that means they cooperate together to attain better overall clustering quality. One way to enable concurrent implementation of the multiple clustering algorithms is by using cooperative clustering. The cooperative clustering model is mainly based on following four components:

1. cooccurred sub-clusters
2. histogram representation of similarities within the sub-clusters
3. cooperative contingency graph
4. coherent merging of sub-clusters

These components are developed to obtain a cooperative model which is capable of clustering data with better quality.

Ensemble clustering: This clustering is based on the idea of combining multiple clusterings of a given dataset to obtain a superior aggregated solution based on aggregation function [4] - [6]. Ensemble clustering techniques are effective in improving the quality. However, the main drawbacks of these techniques are: (1) the excessive computational cost for generating and combining multiple clusterings of the data, and (2) design of a proper cluster ensemble which can address the problems associated with high dimensionality and parameter tuning.

Hybrid Clustering: Hybrid Clustering is another form of combining multiple clusterings. This clustering technique assumes that a set of cascaded clustering algorithms that cooperate together for refining the clustering solutions produced by a former clustering algorithms. However, in hybrid clustering one or more of the clustering algorithms need to stay idle till a former algorithms completes its clustering

which causes a significant waste in the total computational time [7], [8].

The cooperative Clustering (CC) model achieves simultaneous execution of the multiple invoked techniques with no idle time and it obtains clustering solutions with better homogeneity than those of the non-cooperative clustering algorithms. The cooperative clustering model is based on four components. They are Co-occurred sub-clusters, Histogram representation of the pair-wise similarities within sub-clusters, the cooperative contingency graph, and the coherent merging between the set of histograms. These components are developed to obtain an improved cooperative clustering model which is capable of clustering data with better quality than that of the adopted non-cooperative techniques.

Distributed Cooperative Clustering: Centralized clustering usually causes high computational complexity. Distributed clustering therefore achieves a level of speedup that outweighs communication overhead. The distributed clustering can produce either globally or locally optimized clusters. Globally optimized clusters shows the grouping of data across all nodes, as if data from all nodes were pooled into a central location for centralized clustering [20] whereas locally optimized clusters generate a different set of clusters at each node, taking into consideration remote clustering information and data at other nodes. This implies exchanging data between nodes so that certain clusters appear only at specific nodes [21]. If the whole clusters are desired to be in one place rather than fragmented across many nodes then the locally optimized clusters are useful.

2. Cooperative Clustering For Outliers Detection

The problem of discovering objects that do not conform to expected behavior in a given dataset is referred to as outlier detection and such nonconforming objects are known as 'outliers'. There are a variety of techniques have been developed to detect outliers in several research applications such as bioinformatics and data mining [9] - [19]. Current clustering-based outlier detection approaches explore the relation of an outlier to the clusters in data. Traditional clustering-based approaches for detecting outliers are based only on the assumption that outliers either do not belong to any cluster or form very small-sized clusters. A novel clustering-based outlier detection method called Cooperative Clustering Outliers Detection (CCOD) algorithm is proposed and analyzed which provides efficient outlier detection and data clustering capabilities in the presence of outliers [1]. CCOD uses the notion of cooperative clustering towards better discovery of outliers. It is mainly based on three assumptions which are outliers form very small clusters, outliers may exist in large clusters, and outliers reduce the homogeneity of the clustering process. Based on these three assumptions, the outlier detection algorithm first obtains a set of subclusters as an agreement between the multiple clusterings using the cooperative clustering method. A large sub-cluster means strong agreement and a small sub-cluster indicates week agreement. The CCOD involves an iterative identification of possible and candidate outliers of objects in a bottom-up fashion in its following stages. The empirical results show that this method is successful in detecting outliers compared to the traditional clustering-based outlier's detection techniques.

3. Combining Multiple Clustering

Based on the level of cooperation between the clustering algorithms for combining clusterings can be classified into two categories; either they cooperate on the intermediate level or at the end-result level. The examples for cooperation at the end-result level are the ensemble clustering and the hybrid clustering [4] - [8]. In ensemble clustering multiple clusterings of a given dataset X is combined to produce a superior aggregated solution based on aggregation function. This clustering method integrates a collection of "base clusterings" to obtain a more accurate partition of a dataset. Recent ensemble clustering techniques have been shown to be more effective in improving the accuracy and stability of standard clustering algorithms. It can also provide novel, robust, and stable solutions. However, main drawbacks of these techniques are the computational cost of generating and combining multiple clusterings of the data and designing a

proper cluster ensemble that addresses the problems associated with high dimensionality and parameter tuning.

Hybrid clustering is based on the idea of combining together a set of cascaded clustering algorithms for the goal of refining the clustering solutions produced by a former clustering algorithm(s) or to reduce the size of the input representatives to the next level of the cascaded model. Hybrid PDDP-k-means algorithm [8] starts by running the PDDP algorithm [22]. It then enhances the resulting clustering solutions using the k-means algorithm. As one or more of the clustering algorithms stays idle till a former algorithm(s) finishes it clustering, Hybrid clustering violates the synchronous execution of the clustering algorithms at the same time.

3.1. Scalability of the cooperative model

Let us assume that B is the clustering technique that will be added to the cooperative model and B contains c clustering algorithms. If the set of sub-clusters Sb remains the same, then the resulting c+1cooperation is almost the same as the c cooperation. However, if adding B generates a new set of sub-clusters with better homogeneity than the old sub-clusters, then the new set of sub-clusters acts as incremental agreement between the c clustering techniques and the additional approach B. Thus adding the new technique B to the cooperative model was beneficial and it moved the cooperative clustering process into a more homogenous clustering process. The OWSR measure is used to evaluate this homogeneity. In general, increasing the number of algorithms in the cooperative clustering model will increase the number of sub-clustersn_{sb}; the upper bound of number of sub-clusters is k^c, where k is number of clusters and c is number of clustering techniques. Therefore If c is large enough then the number of the generated sub-clusters $n_{sb} \rightarrow n$, which extremely increases the computational complexity of the cooperative model. In this case, each sub-cluster will be considered as a singleton sub-cluster with a maximum of one object and with a similarity ratio of value equals zero then the quality of the sub-clusters will be of a minimum value. Thus after a specific value of c, c*, the cooperative quality decreases rapidly. Then no more techniques can be added to the model after c*.

4. Taxonomies of Clustering Algorithms

Clustering algorithms can be classified using different independent dimensions. Different methodologies, starting points, algorithmic point of view, clustering criteria, and output representations etc. will lead to different taxonomies of clustering algorithms. The different properties of clustering algorithms can be described as given below:

Agglomerative vs. Divisive Clustering: This concept is related to algorithmic structure and operation. An agglomerative approach considers each object as a singleton cluster when it begins, and then at each level it merges clusters together until a stopping criterion is satisfied. This approach is based on bottom-up hierarchical clustering. On the other hand, a divisive method starts with all objects in a single cluster and it performs splitting at each level until a stopping criterion is met and this approach is based on top-down hierarchical clustering.

Monothetic vs. Polythetic Clustering: Both the monothetic and polythetic clusterings are related to the sequential or simultaneous use of features in the clustering algorithm. Most of the clustering algorithms are polythetic; in which all features enter into the computation of distances or similarity functions between objects. The decisions of polythetic Clustering are based on those distances, whereas, a monothetic clustering algorithm uses the features in the clustering algorithm one by one.

Hard vs. Fuzzy Clustering: A hard clustering algorithm allocates each object to a single cluster during its operation and the algorithm outputs a Boolean membership function either 0 or 1. A fuzzy clustering method works by assigning degrees of membership for each input object to each cluster. By assigning each object to the cluster with the largest degree of membership a fuzzy clustering can be converted to a hard clustering.

Distance vs. Density Clustering: A distance-based clustering algorithm assigns an object to a cluster based on its distance from its representative(s) or the cluster, whereas a density-based clustering grows a cluster as long as the density or number of objects in

the neighborhood satisfies some threshold. The distance-based clustering algorithms can typically find only spherical-shaped clusters and it encounters difficulty at discovering clusters of arbitrary shape, whereas density-based clustering algorithm is capable of finding arbitrary shape clusters.

Partitional vs. Hierarchical Clustering: A Partitional clustering algorithm produces a single partition of the data instead of generating a clustering structure such as the dendrogram generated by a hierarchical technique. That is partitional clustering produces flat groups with no hierarchy. Partitional methods have advantages in applications which involve large data sets for which the construction of a dendrogram is computationally prohibitive. A problem encountering in the use of partitional algorithms is the choice of the number of clusters. Hierarchical clustering algorithms are based on the bottom-up approach and Agglomerative Hierarchical Clustering (AHC) algorithms have been commonly employed for software clustering.

Deterministic vs. Stochastic Clustering: This issue is most relevant to the partitional techniques that are designed to optimize a squared error function. Deterministic optimization can be done using traditional techniques in a number of deterministic steps whereas stochastic optimization randomly searches the state space consisting of all possible solutions.

Incremental vs. Non-incremental Clustering: This issue arises when the objects that are set to be clustered are large, and constraints on execution time or memory space need to be taken into account during the design of the clustering algorithm. Incremental clustering algorithms can minimize the number of scans through the objects set and reduce the number of objects examined during execution, or it can reduce the size of data structures used in the operations of the algorithm. Incremental algorithms also do not require the full data set to be available beforehand. It is possible to introduce new data without the need for re-clustering.

Intermediate vs. Original Representation Clustering: When clustering large and high dimensional datasets some clustering algorithms use an intermediate representation for dimensions reduction. This starts with an initial representation, considers each data object and then it modifies the representation. This kind of algorithms use one scan of the dataset and its structure occupies less space than the original representation of the dataset; therefore it may fit in the main memory.

4.1. Clustering Evaluation Criteria

The previous section has reviewed the taxonomy of a number of clustering algorithms which is used to partition the data set based on different clustering criteria. Single or different clustering algorithms that are using different parameters generally result in different sets of clusters. Therefore, the evaluation of clustering is important to compare various clustering results and select the one clustering algorithm that best fits the "true" data distribution. The clustering results of any clustering algorithm should be evaluated using an informative quality measure(s) that reflects the ''goodness'' of the resulting clusters [2]. Cluster validation is the process of assessing the quality and reliability of the cluster sets derived from various clustering processes [1].

Generally, the cluster validity has two aspects; first, both the internal and external the quality of clusters can be measured in terms of homogeneity and separation on the basis of the definition of a cluster: objects within one cluster are similar to each other and they are dissimilar from objects in other clusters. Thus internal quality measures are used to compare different sets of clusters without reference to external knowledge if the data is not previously classified, and the second aspect relies on a given "ground truth" of the clusters. The "ground truth" comes from the domain knowledge, such as known function families of objects, or from other knowledge repositories such as the clinical diagnosis of normal or cancerous tissues for gene expression datasets. Therefore the cluster validation is based on the agreement between clustering results and the "ground truth". Consequently, the evaluation depends upon a prior knowledge about the class labels, i.e. Classification of the data objects. This labeling is used to compare the resulting clusters with the original classification and such measures are known as external quality measures. External quality measures including F-measure, entropy, and purity (used by [23]) are used based on a correct classification [24, 25]. The SI Index [24] can be used

as the internal quality measure, which does not require a prior classification about the objects.

5. Conclusion

The Cooperative clustering (CC) model was developed to improve the clustering solutions over the traditional non-cooperative techniques. The Cooperative Clustering model is based on finding the intersection between the multiple clusterings in terms of a set of sub-clusters. These sub-clusters are represented by similarity histograms. The CC model applies a homogeneous merging procedure on the cooperative contingency graph to attain the same number of clusters by carefully monitoring the pair-wise similarities between objects in the sub-clusters. Based on the study conducted several conclusions can be made; the CC model is scalable in terms of number of clustering algorithms and it provides clustering solutions of better quality. The cooperative clustering is capable of achieving better clustering quality measured by both internal and external quality measures than the non-cooperative traditional clustering algorithms.

6. References

[1] Kashef, R.F., Cooperative clustering model and its applications. Ph.D. thesis,University of Waterloo., 2008.

[2] Kashef, R., Kamel, M.S., Cooperative clustering. Journal of Pattern Recognition43 (6), 2010.

[3] Rashid Naseem, Onaiza Maqbool, Siraj Muhammad., Cooperative clustering for software modularization. The journal of Systems and Software., 2013.

[4] Y. Qian and C. Suen. “Clustering Combination Method”. In International Conference on Pattern Recognition. ICPR 2000, volume 2, pp: 732-735, 2000.

[5] A. Strehl and J. Ghosh. “Cluster Ensembles – Knowledge Reuse Framework for Combining Partitionings”. In conference on Artificial Intelligence (AAAI 2002), pp: 93–98, AAAI/MIT Press, 2002.

[6] H. Ayad, M. Kamel, "Cumulative Voting Consensus Method for Partitions with Variable Number of Clusters," IEEE Transactions on Pattern Analysis and Machine Intelligence, IEEE Computer Society Digital Library. IEEE Computer Society, Volume 30, Issue 1, pp: 160-173, 2008.

[7] Y. Eng, C. Kwoh, and Z. Zhou, "On the Two-Level Hybrid Clustering Algorithm" AISAT04 - International Conference on Artificial Intelligence in Science and Technology, pp: 138-142. 2004.

[8] S. Xu and J. Zhang, “A Hybrid Parallel Web Document Clustering Algorithm and its Performance Study”, Journal of Supercomputing, Vol. 30, Issue 2, pp: 117-131, 2004.

[9] V. Barnett, T. Lewis, Outliers in Statistic Data, John Wiley’s Publisher, NY, 1994.

[10] M. Knorr, and R. Ng, “Algorithms for Mining Distance-Based Outliers in Large Datasets”, Very Large Data Bases Conference Proceedings, pp: 24-27,1998.

[11] F. Angiulli, C. Pizzuti, “Fast Outlier Detection in High Dimensional Spaces”. Proc of PKDD’02, pp: 15-26, 2002.

[12] M. Breunig, H. Kriegel, R. Ng, J. Sander, “LOF: Identifying Density based Local Outliers”, Proc. ACM SIGMOD 2000. Int. Conf. On Management of Data, pp: 93-104, 2000.

[13] K. Yamanishi, J. Takeuchi, G. Williams and P. Milne, “On-Line Unsupervised Outlier Detection Using Finite Mixtures with Discounting Learning Algorithms”. In Proceedings of the 147 International Conference on Knowledge Discovery and Data Mining, Volume 8, Issue 3, pp: 275 - 300, 2004.

[14] A. Arning, R. Agrawal, and P. Raghavan. “A Linear Method for Deviation Detection in Large Databases”. In: Proc of KDD’96, pp: 164-169, 1996.

[15] Z. He, X. Xu and S. Deng**, “**Discovering Cluster-based Local Outliers”, Pattern Recognition. Letters Volume 24, Issues 9-10, , pp: 1641-1650, 2003.

[16] Z. He, X. Xu, and S. Deng,”Squeezer: An Efficient Algorithm for Clustering Categorical Data”. J. Comput. Sci. Technol. 17 5, pp: 611–624. 2002.

[17] S. Ramaswamy, R. Rastogi, and K. Shim, “Efficient Algorithms for Mining Outliers from Large Datasets”, Proceedings of the ACM SIGMOD international conference on Management of data, pp:427-438, 2000.

[18] S. Papadimitriou, H. Kitagawa, P.Gibbons, and C. Faloutsos, "LOCI: Fast Outlier Detection Using the Local Correlation Integral", 19th International Conference on Data Engineering (ICDE03), pp: 315-326, 2003.

[19] D.Hawkins, Identification of Outliers, Chapman and Hall London, 1980.

[20] S. Datta, C. Giannella, and H. Kargupta. "K-means Clustering Over a Large, Dynamic Network". In SIAM International Conference on Data Mining (SDM06), pp: 153-164, 2006.

[21] K. Hammouda and M. Kamel. "Distributed Collaborative Web Document Clustering Using Cluster Keyphrase Summaries". Information Fusion, Special Issue on Web Information Fusion, pp: 465-480, 2008.

[22] D. Boley. "Principal Direction Divisive Partitioning". Data Mining and Knowledge Discovery, 2(4), pp: 325-344, 1998.

[23] R. Kashef, M. Kamel, Enhanced bisecting k-means clustering using inter- mediate cooperation, Pattern Recognition 42 (11) pp: 2557-2569, 2009.

[24] U. Maulik, S. Bandyopadhyay, Performance evaluation of some clustering algorithms and validity indices, IEEE Transactions on Pattern Analysis and Machine Intelligence 24 (12) pp: 1650-1654, 2002.

[25] K. Wu, M. Yang, J. Hsieh, Robust cluster validity indexes, Pattern Recognition 42(11) pp:2541-2550, 2009.