A Survey of Cluster Ensemble

S.Sarumathi
Associate Professor
Department of IT
K.S.Rangasamy College of Technology
Tiruchengode

N.Shanthi
Head of the department
Department of IT
K.S.Rangasamy College of Technology
Tiruchengode

G.Santhiya
PG scholar
Department of IT
K.S.Rangasamy College of Technology
Tiruchengode

ABSTRACT

Cluster ensembles are assortment of individual solutions to a certain clustering crisis which are required to consider in a wide sort of applications. This paper gives the general process of the cluster ensemble and overview of different types of consensus function.

Keywords

Clustering, Cluster ensemble, consensus function

1. INTRODUCTION

Data clustering is one of the essential tools for perceptive structure of a data set. It plays a vital and initial role in data mining, information retrieval and machine learning. The basic goal in cluster analysis is to discover natural groupings of objects in a dataset. The data set sometimes may be in mixed nature that it may consist of both numeric and categorical type of data and differ in their individuality. The traditional clustering algorithms are limited in managing datasets that have categorical attributes.

Due to the differences in their features, in order to group these assorted data, it is good to exploit the clustering ensemble method which uses split and merge approach to solve this problem. Recently, cluster ensembles have emerged as an effective solution that is able to overcome these limitations, and develop the strength as well as the quality of clustering results. Clustering ensembles have appeared as a dominant means for improving both the strength and the stability of unproven classification solutions. Cluster ensemble (CE) is the method to merge numerous jogs of dissimilar clusterings to get a common partition of the original dataset. It has become a primary practice when facing cluster analysis problems, due to its ability for recovering the results of simple clustering algorithms.

2. CLUSTER ENSEMBLE

A cluster ensemble system solves a clustering problem in two steps. The first step takes a data set as input and outputs an ensemble of clustering solutions. The second step takes the cluster ensemble as input and combines the solutions to produce a single clustering as the final output. Figure 1 shows the general process of cluster ensemble, that consists of generating a set of clusterings from the similar dataset and combining them into an ultimate clustering[1]. The objective of this combination process is to recover the quality of individual data clusterings. The intend of combining dissimilar clustering results emerged as an unusual approach for improving the quality of the results of clustering algorithms.

There are two major parts in cluster ensemble

1. Generation mechanisms
2. Consensus functions

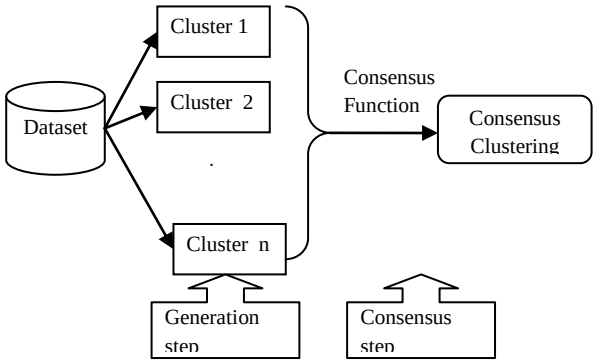

Figure 1 General process of cluster ensemble

2.1 Generation Mechanism

Generation is the first step in clustering ensemble methods, in which the set of clusterings is generated and combined. It generates a collection of clustering solutions i.e., a cluster ensemble. Given a data set of n instances X = {X1,X2, · · · ,Xn}, an ensemble constructor generates a cluster ensemble, represented as $\prod = \{\pi^1,\ldots,\pi^r\}$ where r is the ensemble size (the number of clustering in the ensemble)[8]. Each clustering solution π^i is simply a partition of the data set X into Ki disjoint clusters of instances, represented as $\pi^i = c^i_{1,\ldots}c_k{}^i$

2.2 Consensus Function

The consensus function is the main step in any clustering ensemble algorithm that produces the final data partition or consensus partition , which is the result of any clustering ensemble algorithm, is obtained[2].

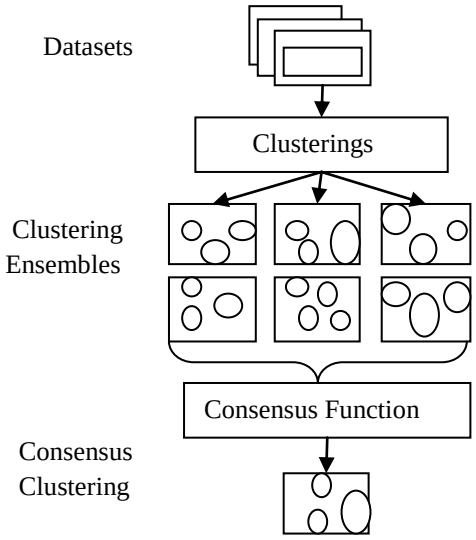

Figure 2 Overview of clustering Ensemble Procedure

There are some types of consensus function such as:

- Co-association based function
- Graph based methods
- Voting approaches
- Mixture model approaches
- Information theory approach
-

2.1.1 Co-association based function

The pair wise similarity method make possible to find the co-occurence relations between the data points[3] and evaluate cluster associations between each of the *N* samples in the dataset to produce an *N*×*N* pairwise similarity matrix i.e. consensus, agreement and co-association matrices, to which a consensus function is applied to obtain the final data separation. An alternative approach to pairwise similarity methods makes use of an N×P binary cluster association matrix (BM) (where P denotes the number of clusters in an ensemble). Given a cluster ensemble $\prod = \{\pi 1,\ldots,\pi M\}$ of a dataset $X = \{x1,\ldots,xN\}$, an *N*×*N* similarity matrix (CO) is constructed as, as

$$CO(xi,xj) = \frac{1}{m}\sum_{m=1}^{M} s_m (x_{i,}y_j)$$

where CO(xi,xj)∈[0,1] represents the similarity measure between samples xi,xj ∈X. In addition, Sm(xi,xj)=1 if Cm(xi)=Cm(xj), and Sm(xi,xj)=0 otherwise and Cm(xi) denotes the cluster label of the m-th clustering to which a sample xi ∈X belongs. co-association matrix (CO) is a matrix of similarity,and any clustering algorithms that is based on similarity can be applied to yield the final partition π∗.

2.2.2 Graph-Based Algorithms

A weighted graph is represented by G = (V;W), where V is a set of vertices and W is a nonnegative and symmetric |V |x |V | similarity matrix characterizing the similarity between each pair of vertices[4]. The contribution to a graph partitioning crisis is a weighted graph G and a number K. To partition a graph into K parts is to find K disjoint clusters of vertices $P = \{P_1, P_2,\ldots P_K\}$ where $U_K P_k = V$. The sum of the weights of these crossed edges is defined as the cut of a partition P: Cut(P;W) = PW(i; j), where vertices i and j do not belong to the same cluster. The general goal of graph partitioning is to find a K way partition that reduces the incise, focus to the constraint that every part should contain roughly the same amount of vertices. There are a number of method for devising graphs from cluster ensemble. They are,

- Cluster Based Similarity Partitioning Algorithm (CSPA)
- Hybrid Graph Partitioning Algorithm (HGPA)
- Meta Clustering Algorithm (MCLA)
- Multiway Spectral Clustering Method (METIS)
- Spectral Graph Patitioning (SPEC)

Cluster Based Similarity Partitioning Algorithm (CSPA)

In the Cluster-based Similarity Partitioning Algorithm (CSPA), from the hypergraph, a similarity matrix n x n (the co-association matrix) is constructed. This can be viewed as the adjacency matrix of a fully connected graph, where the nodes are the elements of the set X and an edge between two objects has an associated weight equal to the number of times the objects are in the same cluster[5]. This is the simplest heuristic and is used in the Cluster-based Similarity Partitioning Algorithm (CSPA). Similarity between two objects is 1 if they are in the same cluster and 0 otherwise. For each clustering, a n x n binary similarity matrix is created. The entry-wise average of **r** such matrices representing the **r** sets of groupings yields an overall similarity matrix. Alternatively, and more concisely, this can be interpreted as using **k** binary cluster membership features and defining similarity as the fraction of clusterings in which two objects are in the same cluster[9]. The entire n x n similarity matrix **s** can be computed in one sparse matrix multiplication

$$S = \frac{1}{r} HH^{+}$$

Hybrid Graph Partitioning Algorithm (HGPA)

The clusters could be represented as hyperedges on a graph whose vertices match to the objects to be clustered, so every hyperedge describes a set of objects belonging to the identical clusters. The crisis of consensus clustering is then reduced to finding the minimum-cut of a hypergraph. The minimum *k*-cut of this hypergraph into *k* components gives the necessary consensus partition[9]. Hypergraph partitioning is NP-hard problem, but competent heuristics to solve the *k* way min-cut partitioning problem are recognized, some with computational complexity on the order of O($|\varepsilon|$), where ε is the number of hyperedges. All hyperedges are considered to have the same weight. Also, all vertices are similarly biased. This contains n_I-way association information, while CSPA only considers pairwise associations. A hyperedge divider that divides the hypergraph into **k** unconnected components of approximately the same size. Equal sizes are obtained by maintaining a vertex imbalance of at most 5% as formulated by the following constraint:

$$K \cdot \max_l \in \{ _{1,\ldots k} \} \frac{n_l}{n} \leq 1.05$$

Meta Clustering Algorithm (MCLA)

In the Meta-Clustering Algorithm (MCLA), The Meta-Clustering Algorithm is based on clustering. It also yields object-wise confidence estimates of cluster membership. First of all the similarity between two clusters is defined in terms of the amount of objects grouped in both, using the Jaccard index. Then, a matrix of similarity between clusters is formed, which represents the adjacency matrix of the graph built considering the clusters as nodes and assigning a weight to the edge between two nodes, equal to the similarity between the clusters. The idea in MCLA is to group and collapse related hyperedges and assign each object to the collapsed hyperedge in which it participates most strongly[9]. The hyperedges that are considered related for the purpose of collapsing are determined by a graph-based clustering of hyperedges. We refer to each cluster of hyperedges as a meta-cluster $C^{(M)}$. crumpling decreases the amount of hyperedges from, $\sum_{q=1}^{r} k^{(q)}$ to K.

Multiway Spectral Clustering Method (METIS)

A multi-level graph partitioning algorithm works by applying one or more stages. Each stage reduces the size of the graph by collapsing edges and vertices, divides the minor graph, then plots back and refines this partition of the inventive graph[7]. The multilevel pattern, consists of three stages: graph coarsening, initial partitioning, and uncoarsening

1. In the graph coarsening stage, a series of successively smaller graphs is derived from the part graph. every consecutive graph is raised from the previous graph by collapsing together a maximal size set of adjacent pairs of vertices.
2. In the initial partitioning stage, a partitioning of the coarsest and hence, smallest, graph is computed using relatively simple approaches.
3. Finally, in uncoarsening stage, the partitioning of the smallest graph is projected to the successively larger graphs by assigning the pairs of vertices that were collapsed together to the same partition as that of their corresponding collapsed vertex.

Spectral Graph Partitioning (SPEC)

Spectral graph partitioning chooses a popular multi-way spectral graph partitioning algorithm, which tries to find to optimize the regulate cut criterion. We submit to this algorithm as SPEC[4]. SPEC can be simply described as follows. Given a graph G = (V;W), it first computes the degree matrix D, which is a diagonal matrix such that D(i,i) = Pj W(i; j). Based on D, it then computes a normalized weight matrix $L = D^{-1}W$ and finds L's K largest eigenvectors u1,u2,…, uK to form matrix U = [u1,…,uK]. The rows of U are then normalize to have unit length. Treating the rows of U as K dimensional embeddings of the vertices of the graph, SPEC produces the final clustering solution by clustering the embedded points using K-means.

2.2.3 Voting Approaches

Voting approach is also called as direct approach or relabeling. The concept of voting used bagging to improve the accuracy of clustering process. Once grouping is completed on a bootstrapped model, the cluster association difficulty is solved using iterative relabeling algorithm. Clustering on each bootstrapped model gives some votes corresponding to each data point and cluster label pair which, in cumulative, decides the final cluster assignment. The main idea is to permute the cluster labels such that best agreement between the labels of two partitions is obtained[5]. All the partitions from the ensemble must be relabeled according to a permanent orientation partition. After an overall reliable relabeling, voting can be practical to determine cluster membership for every object. However, this voting method needs a very large number of clusterings to obtain a consistent result. Computing complexity of their proposed algorithm is O(*k*3).

2.2.4 Mixture Model Approaches

The main assumption is that the labels are modeled as random variables drawn from a probability distribution described as a mixture of multinomial component densities[1]. The objective of consensus clustering is formulated as a maximum likelihood estimation problem. To find the best fitting mixture density for a given data *Y* we must maximize the likelihood function with respect to the unknown parameters Θ.

$$\log(\Theta|Y) = \log \prod_{i=1}^{N} P(y_i|\ \Theta) = \sum_{i=1}^{N} \log \sum_{m=1}^{M} a_m P_m(y_i|\ \Theta_m)$$

The probing of the consensus partition is devised as a problem of maximal likelihood evaluation:

$$\Theta^* = \arg_{\Theta} \text{Max } \log L(\Theta|Y)$$

2.2.5 Information Theory Approach

The objective function for a clustering ensemble can be originated as the mutual information (MI) between the experimental probability distribution of labels in the consensus partition and the labels in the ensemble[6]. Quadratic Mutual Information (QMI) or feature based approach can be effectively maximized by the *K*-means algorithm in the space of particularly altered cluster labels of given ensemble. The collection of *L* features can be regarded as an "intermediate feature space" and another clustering algorithm can be run on it.

The below table shows the comparison among different approaches of consensus function among the computational complexity, scalability, robustness and ease of implementation[10].

Consens-us Function	**Computatio-nal Complexity**	**Scalability**	**Robustnes**	**Ease of Implementati-on**
Mixture Models	$O(K^3)$	High	Low	Easy to Implement
Voting Based Approach	$O(K^3)$	High	High	Easy to Implement
Informati-on Theory Approach	$O(K^3)$	Low	High	Not easy to Implement
Co-Associati-on Based Approach	$O(N^2)$	High	High	Difficult to Implement
Hypergra-ph Based Approach	$O(N^3)$	High	High	Difficult to Implement

3. Conclusion

Clustering ensemble is a foremost technique emerged and acts as a major keystone for overcoming the drawbacks of individual clustering consequences. Hence In this paper, we survey some of the major clustering ensemble approaches captivating into report their theoretical description and the mathematical computation, used by all means.The paper describes the general process of cluster ensemble and different types of consensus function. From the above discussion, the voting based approach is pertained to be the more suitable mechanism due to its high quality accurate measure with efficient relabeling technique when compared to other consensus function approaches that produces the consensus partition. Our future research effort will focus on achieving better consensus results in Mixed Numerical and Categorical datasets using Voting based approach.

4. References

[1] Sandro vega-pons and jose ruiz-shulcloper, "a survey of clustering ensemble algorithms" international journal of pattern recognition and artificial intelligence vol. 25, no. 3 (2011).

[2] Javad Azimi, Paul Cull and Xiaoli Fern, "Clustering Ensembles Using Ants Algorithm" EECS Department, Oregon State University, Corvallis, Oregon, 97330, USA.

[3] Natthakan Iam-on, Tossapon Boongoen and Simon Garrett," LCE: a link-based cluster ensemble method for improved gene expression data analysis" Vol. 26 no. 12(2010).

[4] Xiaoli Zhang Fern and Carla E. Brodley," Solving Cluster Ensemble Problems by Bipartite Graph Partitioning" International Conference on Machine Learning, Banff, Canada, (2004).

[5] Joydeep Ghosh and Ayan Acharya," Cluster ensembles" Volume 1, Jul y /August (2011).

[6] Reza Ghaemi , Md. Nasir Sulaiman , Hamidah Ibrahim , Norwati Mustapha "A Survey: Clustering Ensembles Techniques" (2009).

[7] George Karypis," METIS-A Software Package for Partitioning Unstructured Graphs, PartitioningMeshes, and Computing Fill-Reducing Orderings of Sparse Matrices" August 4, (2011).

[8] Xiaoli Z. Fern and Carla E. Brodley "Cluster Ensembles for High Dimensional Clustering: An Empirical Study"(2004)

[9] Alexander Strehl and Joydeep Ghosh"Cluster Ensembles A Knowledge Reuse Framework for Combining Multiple Partitions"(2002)

[10] Ashraf Mohammed Iqbal, Abidalrahman Moh'd, and Zahoor Ali Khan" Semi- supervised Clustering Ensemble by Voting"