Chapter – 3

A Relational Graph Based Approach using Multi-Attribute Closure Measure for Categorical Data Clustering
CHAPTER - 3

A RELATIONAL GRAPH BASED APPROACH USING MULTI-ATTRIBUTE CLOSURE MEASURE FOR CATEGORICAL DATA CLUSTERING

3.1. INTRODUCTION

As the technology development rising at each moment of life the people started using various ways to find information from the knowledge base. Clustering is one of the machine learning method to group similar information as a subset, so that to facilitate the retrieval of information as easier one and to provide good support for the learner. There has been various standard methods in the domain of information mining and grouping. K-Means a popular method to group numerical data points which takes set of parameters as an argument with the input data set and returns a number of subset of data points as mentioned in the parameter.

The k-means algorithm takes argument as how many number of clusters has to be formed and it computes distance measure using Euclidean distance or any of the method to compute the distance between data points. The process computing distance and shifting the data point from one group to another will be iterated until there is no data point has to be shifted from a cluster to another. This process will be controlled by an input parameter specified by number of iteration. The quality of clustering is depend on the method of distance computing and number of iterations. Mainly k-means will be useful for single attribute based clustering approaches, while this will not be efficient and useful in case of multi attribute based clustering approach.

Categorical data is one which contains as many numbers of attributes and all are discrete without any ordering. For example the attributes of normal human being
are HD={sex, marital} which has discrete values of sex={male, female} and marital={married, unmarried}. Like this any data point can have as many numbers of attributes but clustering these data points are difficult in nature.

The problem of categorical data clustering can be handled with the help of relational graph. The relational graph is one which has number of edges towards the attributes of data points. For any class, the graph has number of edges which represents that the number of dimensions are related to the data points and number of dimensions are similar to the input data point.

![Figure 3.1: Example scenario of relational graph](image)

The Figure 3.1 shows the example scenario of relational graph. The Figure has two graphs, where the youngster graph represents a group which has two vertices namely sex as men and age with threshold 50. In another graph young black women, there are three dimensions like sex, age and color.

In this case, for any given data point Di, if the dimension of sex has the value Men, with the age below 50 then the data point is assigned to the class youngster. On
the other side, if the sex is women and age below 30 with the color as black then it
will be assigned to the class young black women.

However, identifying the class of data point with the least dimensional data
point will be easier but when the dimension grows then it will be difficult to identify
them. To solve this issue, this chapter discusses the relational graph approach which
generates relational graph for each class and for any given input data point, the
method generates a graph and computes similarity with each dimensional value. If the
dimension is close to the input data point value, then the method generates an edge to
the vertex. It shows the relation it has with the dimension. Similarly, the method
generates number of edges to the vertices according to the closure of the dimensions.
Finally the total relational factor is computed which represent the class to which the
input data point belongs to.

Many number of algorithms have been introduced for clustering categorical
data, each has its own strengths and weaknesses. For a particular data set, different
algorithms, or even the same algorithm with different parameters, usually provide
distinct solutions. Therefore, it is difficult for users to decide which algorithm would
be the proper alternative for a given set of data. Recently, cluster ensembles have
emerged as an effective solution that is able to overcome these limitations, and
improve the robustness as well as the quality of clustering results. The main objective
of cluster ensembles is to combine different clustering decisions in such a way as to
achieve accuracy superior to that of any individual clustering. For example the
feature-based approach transforms the problem of cluster ensembles to clustering
categorical data (i.e., cluster labels), the direct approach that finds the final partition
through relabeling the base clustering results, graph-based algorithms that employ a
graph partitioning methodology and the pair wise-similarity approach that makes use of co-occurrence relations between data points.

3.2. METHODS EXPLORED

Many approaches have been discussed in the past for clustering of categorical data, few of them has been explored in this chapter to find the prose and corners of the methods proposed.

A Robust Clustering Algorithm for Categorical Attribute [1] makes use of a link graph, in which nodes and links represent data points (or tuples) and their similarity, respectively. Two tuples are similar if they shared a large number of attribute values. Note that the link connecting two nodes is included only when the corresponding similarity exceeds a user-defined threshold. With tuples being initially regarded as singleton clusters, ROCK merges clusters in an agglomerative hierarchical fashion, while optimizing a cluster quality that is defined in terms of the number of links across clusters. Note that the graph models used by ROCK and LCE are dissimilar — the graph of data points and that of attribute values (or clusters), respectively. Since the number of data points is normally greater than that of attribute values, ROCK is less efficient than LCE. As a result, it is unsuitable for large datasets [2]. Also, the selection of a “smooth function” that is used to estimate a cluster quality is a delicate and difficult task for average users [4].

CACTUS [3] also relies on the co-occurrence among attribute values. In essence, two attribute values are strongly connected if their support (i.e., the proportion of tuples in which the values co-occur) exceeds a prespecified value. By extending this concept to all attributes, CACTUS searches for the “distinguishing
sets,” which are attribute value sets that uniquely occur within only one cluster. These sets correspond to cluster projections that can be combined to formulate the final clusters.

Consensus Clustering [8], the cluster the data points using three consensus like Iterative Voting Consensus, Iterating Probability Consensus and Iterative pairwise consensus. This method performs the three steps to form the final cluster. This is and cluster ensample approach which is not using the underlying data features of the data points.

In [10] an weighted ensample approach is discussed, it proposes two algorithms namely WSPA and WBPA provide as output a partition of the data into k clusters, with no information regarding feature relevance for each of the clusters. Clustering ensemble algorithm (WSBPA) that provides weighted clusters in output. This technique advances the WBPA method by adding to the final partition weighted features associated with each cluster. By assigning a value to each dimension, WSBPA captures the local relevance of features within each cluster. Thus, the structure of the output provided by a single run of LAC is preserved.

Combining multiple clustering using evidence accumulation [11], produces a clustering ensemble - a set of object partitions from a data set (n objects or patterns in d dimensions). To produce the cluster ensample different ways are used as:

(1)- applying different clustering algorithms, and (2)- applying the same clustering algorithm with different values of parameters or initializations. Further, combinations of different data representations (feature spaces) and clustering algorithms can also provide a multitude of significantly different data partitionings. They propose a simple framework for extracting a consistent clustering, given the
various partitions in a clustering ensemble. According to the EAC concept, each partition is viewed as an independent evidence of data organization, individual data partitions being combined, based on a voting mechanism, to generate a new \( n \times n \) similarity matrix between the \( n \) patterns. The final data partition of the \( n \) patterns is obtained by applying a hierarchical agglomerative clustering algorithm on this matrix.

Refining Pairwise Similarity Matrix for Cluster Ensemble Problem with Cluster Relations\[12\], used two new similarity matrices, which are empirically evaluated and compared against the standard co-association matrix on six datasets (both artificial and real data) using four different combination methods and six clustering validity criteria. In all cases, the results suggest the new link-based similarity matrices are able to extract efficiently the information embedded in the input clusterings, and regularly suggest higher clustering quality in comparison to their competitor.

Link-Based Cluster Ensemble Approach [13], proposed a novel technique for Categorical Data Clustering. It is more efficient than the former model, where a BM-like matrix is used to represent the ensemble information. The focus has shifted from revealing the similarity among data points to estimating those between clusters. A new link-based algorithm has been specifically proposed to generate such measures in an accurate, inexpensive manner. It uses weighted graph to identify the links between the clusters.

Most of the proposed methodologies are prevalent in nature suffers with initial set of clustering i.e. identifying the ensamples. The ensamples are generated based on the method of existing algorithm like k-means or some other algorithm. They perform clustering iteratively in the given ensamples. But all the methods are suffers with
identifying the number of clusters has to be formed. A new graph based method for clustering has been proposed of categorical data which could identify and form number of clusters dynamically.

A set of \( N \) data points \( X = \{x_1; x_2; \ldots; x_n\} \) and a set of Clustering \( C = \{C_1, C_2, \ldots, C_n\} \) of the data points in \( X \) are given. Each clustering \( C_i \) is a mapping from \( X \) to \( \{1, \ldots, n_x\} \) where \( n_x \) is the number of clusters in \( C \).

### 3.3. RELATIONAL GRAPH BASED CATEGORICAL DATA CLUSTERING

Conventional clustering approaches suffer with the scalability of number of attributes based on which the clustering is performed. There are approaches to cluster data points with multiple attributes but suffers with overlapping and multiple iteration needed to perform clustering, also the measure computed for the variation of data points between cluster also will not be effective when doing with multiple attributes. To overcome the problem a new graph based approach has been proposed which represents the relation between the data points and clusters. The relational graph consists of various vertices and edges, each vertex represent a data point. There will be an edge between two different edges only if there is a multi-attribute closure between them. The attribute closure, has been computed using all the attributes of the data points. The threshold method to select the data point has closure to other one and the value of threshold is set based on number of attributes the data point has. The proposed method produces good results compare to other approaches discussed in this era and we have evaluated the proposed method with various data sets.

The relational graph based categorical data clustering consists of following steps:

1. Initial Clustering
2. Relational Graph Based Clustering
3. Cluster Validation.
The Figure 3.2 shows the architecture of relational graph based categorical data clustering and shows the functional components in detail.

### 3.3.1. Initial Clustering

The proposed system produces initial clustering based on multi attribute variance method, generated variance and number cluster will be given and could be modified according to the requirement of the user. Given N data samples, we compute attribute based variance $Dev_a$ for each attribute of $A\{a_1, a_2, ..., a_n\}$. Once we compute the attribute variance then number of possible clustering is computed using...
the cumulative attribute variance and that many number of cluster will be formed. The same variance will be used to identify the cluster of data point $D_{d_i}$.

Figure 3.3: Flow chart of Initial Clustering

The Figure 3.3 shows the flow chart of initial clustering and shows the stage by stage process in detail.
Pseudo Code of Initial Clustering:

Input: Data Set D

Output: Initial Cluster C

Read input data set D.

identify number of data samples N.

initialize sd, AF, AT, csd, psd.

for each attribute Ai of d_i from D

    SA = sort(A(D)).

    For each SA_i

        cumulative variance Sdn = \mu (\sum \sigma^2 (A_0, A_1, A_n).)

        Csd=sdn; // current variance.

        Psd = sdn. // previous variance.

        If csd>psd then

            sd(i) = sdn .

            AF(i)=0;

            AT(i)= n.

            Sdn=0;

            I=n.

        End

    End

Nc=\Delta(sd)

create cluster C={c1,c2..c_{Nc}};

Assign labels to data points.

Stop

The above discussed algorithm generates the initial cluster which will be used to perform the final clustering and validate.
3.3.2. Relational Graph Clustering

The relational graph is constructed using the multi attribute closure (MAC) measure. The MAC is computed as the sum of number of attribute values and the average of them. This represents the overall similarity of the data points and clusters. With the computed MAC the graph is constructed. Given D data points, a relation graph \( G=(V,\text{MAC}) \) can be constructed where \( v \) is the set of vertices which denotes the set of clusters and MAC denotes the multiple attribute closure which represents the closeness of the clusters.

![Flow chart of Relational Graph Clustering](image)

**Figure 3.4: Flow chart of Relational Graph Clustering**

82
The Graph 3.4 shows the flow chart of relational graph clustering and shows the stage by stage process in detail.

Pseudo Code of Relational Graph Clustering:

**Input:** Initial Cluster C

**Output:** Final Cluster C.

start

read initial clusters C.

create graph G.

for each cluster \( C_i \) from C

  Construct a \( G(V) \) vertex in G.

  Extract data points \( D \) from \( C_i \)

  Extract data points \( D_1 \) from \( C_{i+1} \)

  Compute \( \text{MAC} = \frac{\sum \phi(D(A_i)-D_1(A_i))}{\Omega(D)}. \)

  \( \phi \)- sum of all equivalence of attributes of data points of clusters.

  \( \Omega \) - total number of data points from both the clusters.

End

if \( \text{MAC} > C_{th} \)

  Perform clustering between two clusters.

End

Stop.

The above discussed algorithm computes the multi attribute closure using the relational graph being generated at the previous stage and generates the clusters.
3.3.3. Cluster Validation

The validation of the cluster is performed by giving new inputs to the cluster algorithm. The algorithm is evaluated by the cluster label assigned to the data point and verified with the feature of the data points. This procedure is performed iteratively to identify the overlapping members of the cluster.

3.4. CONCLUSION

The relational graph based multi attribute closure based clustering produced good results. The multi attribute closure has affected the quality of clustering very effectively other than previous measures used by other algorithms. The attribute closure measure represent the closeness of the data points and their attribute values, so that the proposed method has done the clustering effectively and also the evaluation results shows the effectiveness of the proposed system. The proposed methodology can still improved with few other closure measures.