Generation of Multiple Clustering Views

Sondawale Harsha A. **Prof. N. M. Shahane**

Abstract — **Clustering is a popular for exploratory data analysis, data representation and data summarization. Traditional clustering algorithms only find single clustering solution which is not sufficient for the analysis of the data. Complex data can be grouped in many different ways. This is true in case of high-dimensional setting where different subspaces give different possible grouping of the data. The different views of the data and the use of relationship between these views are helpful to solve the problem of clustering. For this purpose an approach is introduced that finds the multiple clustering views.**

Keywords— **Brent's method, kernel method, multi-view clustering, spectral clustering**

I. INTRODUCTION

Clustering is nothing but the task of grouping objects in the same group or cluster depending on some similarity. The important task in clustering is to find the essential grouping. Cluster analysis divides data into groups or clusters that are meaningful and useful. Clustering is helpful for exploratory data analysis. The goal of exploratory data analysis is to find structure and interesting patterns in data or to extract information from data. But many clustering algorithms provide single clustering solution which is not sufficient for analysis of data. Complex data can be represented in various ways for many purposes. In high dimensional data, different structure of the data may be shown by different feature subspaces then in such a case why to depend or provide a single clustering view while many other alternative clustering views might be helpful for various other purposes [6]. It may be found that single clustering is not essential and helpful or actionable so there is a need to find the alternative solution. Most of the clustering algorithms provide only single partitioning or clustering of the data which is insufficient for the analysis of data. Multiple clustering views of data are useful for different reasons.

Data are collected from different domains or also obtained from different feature extractors and demonstrate heterogeneous properties, because variables of data can be partitioned into groups. Each variable group is referred to as a particular view or multiple views. In subspace learning the main aim is to obtain latent subspaces shared by different views. The aim of subspace clustering is to detect cluster in any subspace projection of a high dimensional space.

The proposed approach of finding multiple clustering or multi-view clustering is based on spectral clustering [3]. Spectral clustering is most widely used clustering algorithms. It is simple to implement and outperform the traditional clustering algorithm. Spectral clustering is developed from spectral graph theory. Spectral clustering is used to obtain normalized minimum cut of the graph. It has various advantages. It captures a flexible notion of cluster shape and it is not sensitive to outliers or shape of clusters.

II. LITERATURE SURVEY

Although the literature on clustering is huge, there has been relatively not much attention given for finding multiple non-redundant clustering. Multiple alternative clustering solutions can be generated by two ways. One is to find multiple solutions simultaneously and other is to find alternative solutions iteratively. Gondek and Hofmann [4] suggest information-theoretic framework that makes use of the concept of conditional mutual information. In this the important problem of non-redundant data clustering is also investigated based on the idea of maximizing conditional mutual information relative to given background knowledge. This approach is dependent on distributed assumption. Bae and Bailey [5] utilize "cannot-link constraint" and agglomerative clustering to find alternative clustering. Cui et al.[6],[7], finds different clustering views by clustering the subspace orthogonal to the clustering solution found in previous iterations without making use of specific clustering algorithm. CAMI [9] simultaneously discovers two disparate clustering by optimizing cluster quality, quantifying these criteria by maximizing the mutual likelihood of Gaussian mixture models and minimizing mutual information between them. The method described in [8] is based on k-means and CAMI [9] both are limited to convex clusters. There are many related work based on subspace clustering. But research in above papers they did not use spectral clustering. The aim of subspace clustering is to find the clusters which are hidden in high dimensional space. There are two ways of subspace clustering based on search strategy that are top-down algorithm and bottom-up approach. Top-down algorithms find first clustering in full set of dimension and compute the subspaces of each cluster. While in case of bottom-up approach, it finds dense region in low dimension spaces and combine them to form clusters [10]. Several approaches have been used for finding multiple clustering views. But it has some boundaries. To overcome these

limitations and drawbacks is the motivation behind this research.

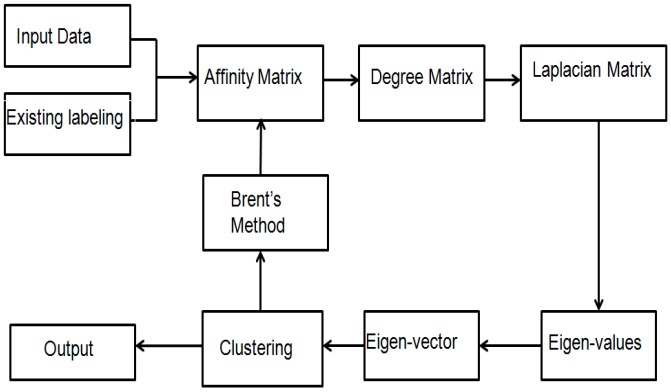

Fig.1. Block diagram of proposed system

III. PROPOSED WORK

A. *Proposed System:*

The clustering solution provided by many clustering algorithm is not what the user is interested in or what the user want. In some cases, experts have prior knowledge on groupings they are not interested in, and would like to find whether the data provide other interesting non-redundant structures they have not yet discovered. When data is multi-faceted, there may include multiple clustering interpretations and different feature subsets may be relevant to each interpretation.

The approach of finding multiple clustering solutions or views is based on spectral clustering. Spectral clustering is one of the most popular clustering algorithms. It is simple to implement and can be solved efficiently. Spectral clustering technique has been developed from spectral graph theory. An advantage of spectral clustering is the ability to form clusters of arbitrary shapes because the graph are fully connected and do not assume any underlying model for any cluster. Hence any data point may be considered connected with any other point.

There are two modes of discovering multiple alternative clustering views – simultaneous or iterative. But our main focus is on iterative discovery of clustering views. Sometimes, when one clustering solution is given, users want to find another non-redundant clustering solution. Approach in this project can solve that problem. This approach allows users to discover one clustering solution alternative to clustering solutions that have been previously provided.

Spectral clustering depends on the Eigen structure of a similarity matrix. In spectral clustering, clusters are formed by dividing data points using similarity matrix. It has three main stages [12], which are pre-processing, spectral mapping and post mapping. Construction of similarity matrix is performed through pre-processing, spectral mapping deals with the construction of Eigen vectors for the similarity matrix and grouping of the data points are performed by post processing.

The figure 1 shows block diagram for finding alternative clustering views. It is based on 3-rules for basic spectral clustering i.e. building the affinity matrix (similarity matrix), determination of eigen-values and eigen-vectors of the matrix and use of these to perform clustering.

The input to the system is n data samples denoted by x_1, x_2,..,x_n. Also the input given to the system is existing labeling matrix $Y = [Y_0,\ldots,Y_{t-1}]$ which is of size n by $\sum c_j$ (j=0 to t-1) , where Y_j is the cluster labeling matrix and c_j is the number of clusters per iteration. If P_0 is the current partitioning or clustering solution denoted by a set of clusters $C_1,\ldots, C_c$ where c is the number of clusters then cluster labeling matrix for P_0 is defined as Y_0 of size n by c where n is the number of data samples and c is the number of clusters. If data point xi belongs to cluster j in P_0 then y_{ij} = 1 otherwise it is 0. Similarly at iteration t, P_j (j=t-1,…,0) is previously found partitioning or clustering solution, at this point all these clustering are represented by augmenting all existing cluster labeling matrices in a single matrix $Y = [Y_0,\ldots,Y_{t-1}]$. Thus in each iteration, Y is given as an input to find alternative clustering solution U of size n by c.

1) Affinity Matrix: Before applying spectral clustering there is need to find the pair-wise similarities between data points. For a set of n data samples $\{x_1, x_2,.., x_n\}$ with each xi is a column vector. A set of similarity is given as $\{k_{ij}\}$ between all pairs xi and x_j. The similarity matrix or affinity matrix K of size n by n is obtained from Gaussian Kernel function:

$$k(x_i,x_j) = \exp(-\|x_i-x_j\|^2 /2\sigma^2)$$

where σ is scaling parameter. The σ determines the width of the Gaussian kernel. It is also called as standard deviation, and the square of it i.e. σ2, is the variance. The value of σ is always positive i.e. $\sigma > 0$.

2) Degree Matrix: From the similarity or affinity matrix the diagonal matrix is calculated as: $\sum k_{ij}$ (j=1 to n).

3) Laplacian Matrix: The next step is to calculate the normalize graph Laplacian i.e. $L = I- D^{-1/2} K D^{-1/2}$ where D is the diagonal matrix of size n by n where the diagonal elements are the degree d_i and I is the identity matrix of size n by n.

4) Eigen-Values and Eigen-Vector: Once the Laplacian matrix is calculated the eigen-values are generated as det(L-

λI) = 0 and for top c eigen-values eigen-vector is calculated. The solution is to set column of matrix U equal to the c eigenvectors corresponding to the largest c eigenvalues of normalized similarity matrix $D^{-1/2}$ K $D^{-1/2}$. This gives the spectral embedding. The discrete partitioning of the data is obtained based on this embedding by renormalizing each row of U to have unit length which is calculated as $Y_{ij} = U_{ij} / (\sum_j U_{ij}^2)^{1/2}$.

5) Clustering: Treating each row of Y as a point, cluster them into number of clusters through k-means algorithm. Finally the original point xi is assigned to cluster j if and only if row i of the matrix Y was assigned to cluster j.

6) Brent's Method: Finally the improvement is made to the spectral clustering algorithm. This algorithm uses a distance matrix with a scaling parameter σ to calculate affinity matrix values, which later can be used to find eigenvectors. The improvement is implemented by the Brent's Method to modify the scaling parameter. The new scaling parameter will give us a better affinity matrix by bringing closer similar points and distancing the not similar ones. The Brent's method is used to minimize the following function.

$$f(\sigma) = \sum_{(si,sj)\epsilon S} (1- \exp(-\| s_i - s_j \| / 2\sigma^2))^2 - \log (\sum_{(si,sj)\epsilon D} (1- \exp(-\| s_i - s_j \| / 2\sigma^2))) \quad (2)$$

S is the set of pairs of points lying in the same cluster and D is the set of pairs of points lying in different clusters. The purpose of this work is to improve the spectral clustering algorithm by fine tuning the scaling parameter σ. Computation of the function will be done at each step of the clusterization yielded by spectral clustering algorithm.

IV. Experimental Setup

A. Dataset:

To get a better understanding of this method and test its applicability, the experiment is performed on synthetic datasets. The synthetic data is generated from six features with 600 instances. This synthetic data contain synthetically generated control charts. This dataset has 600 examples of control charts synthetically generated by the process in Alcock and Manolopoulos (1999). It contains six different classes of control charts.

Experiment on Text Data: The text data consist of 129,000 abstracts from year 1990-2003 describing NSF awards for basic research [13]. In this experiment the subset of this dataset is used. Here the word is represented by each feature and frequency of the word in the text instance is the feature value.

A. Performance Measure:

The challenging problem of clustering is the evaluation of clustering result. Here two types of criteria are used through which the quality of cluster is measured. First one is the external criteria and second one is internal criteria. The external criteria measure the similarity between the alternative clustering solution and the second labelling. If the data set has two labelling then the second labelling is considered as the alternative labelling. The normalized mutual information (NMI) is used for external measure. Suppose U is the alternative clustering and L is the known labels,

$$NMI(L,U) = \frac{I(L,U)}{\sqrt{H(L)H(U)}}$$

where, I(L,U) is the mutual information between L and U and H(L) and H(U) are the entropies of L and U respectively. The clustering and labels are more similar if NMI value is higher.

An internal criterion is used to measure quality. There are two standards to measure internal criteria: mean-squared-error(MSE) and Dunn index(DI). MSE measures the error of instances to its corresponding cluster centroid.

MSE = $1/n \sum_{j=1}^{c} \sum_{x\epsilon Cj} \| x - \mu_j \|^2$, where n is the number of instances and μ_j is the centroid of cluster C_j. Lower value of MSE gives the better cluster quality.

The Dunn index is a ratio of the between cluster separation normalized by the within-cluster distance. For clustering C = $\{c_1,\ldots, c_k\}$, where $\delta : c \times c \rightarrow IR^+$ is cluster to cluster distance and $\Delta : c \rightarrow IR^+$ is cluster diameter measure, DI(c) = $\min_{i\neq j} \{\delta(c_i,c_j)\}/ \max_{1<=l<=k} \{\Delta(c_l)\}$. The quality of cluster is higher if the value of this index is higher.

B. Results:

Experiments on NSF Award Text Data:

The dataset from UCI KDD repository consists of 129,000 abstracts from year 1990 to 2003 describing NSF awards for basic research. For each text instance, bag-of-word data file is extracted. Also, a list of words is provided for indexing the bag-of-word data. The subset of dataset is used to perform the experiment.

The affinity matrix is calculated from input data which is shown in figure. From the affinity matrix degree matrix and Laplacian matrix is generated. The eigen-values and eigen-vector is calculated from Laplacian matrix which gives the spectral embedding. Based on this embedding the discrete partitioning of the data is obtained by renormalizing each row to have unit length, this gives the normalized eigen-vector shown in figure 5. The clusters are formed my applying k-means clustering algorithm on normalized eigen-

vector. Finally the original point x_i is assigned to cluster j if and only if row i of the normalized eigen-vector was assigned to cluster j. Here different cluster gives different clustering solution or views.

	A	B	C	D	E	F	G	H	I	J
1	0	0.885738	0.895569	0.861188	0.888624	0.914537	0.909916	0.902477	0.907923	0.898674
2	0.885738	0	0.890091	0.871493	0.87039	0.905567	0.904133	0.901264	0.885819	0.904314
3	0.895569	0.890091	0	0.876764	0.887768	0.917324	0.909601	0.906716	0.894611	0.916288
4	0.861188	0.871493	0.876764	0	0.862527	0.897385	0.895964	0.893122	0.877816	0.896144
5	0.888624	0.87039	0.887768	0.862527	0	0.904644	0.904223	0.907306	0.876705	0.903705
6	0.914537	0.905567	0.917324	0.897385	0.904644	0	0.943225	0.938633	0.928956	0.937958
7	0.909916	0.904133	0.909601	0.895964	0.904223	0.943225	0	0.944407	0.916033	0.939828
8	0.902477	0.901264	0.906716	0.893122	0.907306	0.938633	0.944407	0	0.913591	0.932227
9	0.907923	0.885819	0.894611	0.877816	0.876705	0.928956	0.916033	0.913591	0	0.91582
10	0.898674	0.904314	0.916288	0.896144	0.903705	0.937958	0.939828	0.932227	0.91582	0

Fig.2. Affinity Matrix

	A	B	C	D	E	F	G	H	I	J
1	8.064646	0	0	0	0	0	0	0	0	0
2	0	8.01881	0	0	0	0	0	0	0	0
3	0	0	8.094732	0	0	0	0	0	0	0
4	0	0	0	7.932402	0	0	0	0	0	0
5	0	0	0	0	8.005891	0	0	0	0	0
6	0	0	0	0	0	8.288229	0	0	0	0
7	0	0	0	0	0	0	8.267331	0	0	0
8	0	0	0	0	0	0	0	8.239744	0	0
9	0	0	0	0	0	0	0	0	8.117272	0
10	0	0	0	0	0	0	0	0	0	8.244958

Fig.3. Diagonal Matrix

	A	B	C	D	E	F	G	H	I	J
1	-0.315	-0.31411	-0.31559	-0.31241	-0.31385	-0.31934	-0.31894	-0.31841	-0.31603	-0.31851
2	-0.55721	0.126998	-0.06174	0.731948	-0.26454	-0.03352	0.050502	0.057007	-0.19	0.144316
3	-0.28031	0.064555	0.118023	-0.02658	0.175188	-0.18145	0.662238	-0.42921	0.275019	-0.37496

Fig.4. Eigen-Vectors

	A	B	C
1	-0.4508	-0.79741	-0.40114
2	-0.91071	0.368211	0.187167
3	-0.92131	-0.18024	0.344543
4	-0.39234	0.919214	-0.03337
5	-0.70325	-0.59275	0.392541
6	-0.86585	-0.09088	-0.49198
7	-0.43289	0.068545	0.898839
8	-0.59244	0.106069	-0.7986
9	-0.68701	-0.41303	0.597853
10	-0.62123	0.281479	-0.73133

Fig.5. Normalized Eigen-Vector

Cluster	No Of Records	Centroid
Cluster1	3	-0.742450952749089, -0.17336121348048...
Cluster2	4	-0.686112746654998, -0.27936699067227...
Cluster3	3	-0.535335368502525, 0.435587523117988...

Fig.6. Clustering Result

CONCLUSION

Many clustering algorithm finds single clustering view or solution which is not sufficient for the analysis of data. Different clustering solutions may be relevant for different purposes. Features relevant to one clustering interpretation may be different from the ones relevant for an alternative interpretation or view of the data. This method gives multiple clustering solutions/views.

ACKNOWLEDGMENT

I would like to express my special thank to all those people who have helped me to complete this work. I am very grateful to my guide, Prof. N. M. Shahane, Associate Professor, Department of Computer Engineering, K. K. W. I. E. E. R., Nashik, for his guidance, encouragement and the interest shown in this project. He has continuously helped and encouraged me in my work.

REFERENCES

[1] H. Sondawale, N. Shahane, " Multiple Clustering Views for Data Analysis". International Journal of Application or Innovation in Engineering and Management, vol. 3, issue 10, October 2014.

[2] H. Sondawale, N. Shahane, " Multiple Clustering Views for Data Analysis using Spectral Clustering". c-PGCON 2015.

[3] Donglin Niu, Jennifer G. Dy, Michael I. Jordan, "Iterative Discovery of Multiple Alternative Clustering Views". IEEE transaction on pattern analysis and machine intelligence, vol. 36, no.7, July 2014.

[4] D. Gondek and T. Hofmann, "Non-Redundant Data Clustering," Proc. IEEE Int'l Conf. Data Mining, pp. 75-82, 2004.

[5] E. Bae and J. Bailey, "COALA: A Novel Approach for the Extrac-tion of an Alternate Clustering of High Quality and High Dissim-ilarity," Proc. IEEE Int'l Conf. Data Mining, pp. 53-62, 2006.

[6] Y. Cui, X.Z. Fern, and J. Dy, "Non-Redundant Multi-View Clus-tering via Orthogonalization," Proc. Seventh IEEE Conf. Data Min-ing (ICDM '07), pp. 133-142, 2007.

[7] Y. Cui, X.Z. Fern, and J.G. Dy, "Learning Multiple Nonredun-dant Clusterings," ACM Trans. on Knowledge Discovery from Data, vol. 4, no. 3, Article 15, 2010.

[8] P. Jain, R. Meka, and I.S. Dhillon, "Simultaneous Unsupervised Learing of Disparate Clustering," Proc. SIAM Int'l Conf. Data Min-ing, pp. 858-869, 2008.

[9] X.H. Dang and J. Bailey, "Generation of Alternative Clusterings Using the CAMI Approach," Proc. SIAM Int'l Conf. Data Mining, pp. 118-129, 2010.

[10] L. Parsons, E. Haque, and H. Liu, "Subspace Clustering for High Dimensional Data: A Review," ACM SIGKDD Explorations News-letter, vol. 6, no. 1, pp. 90-105, 2004.

[11] N. Tishby, F.C. Pereira, and W. Bialek: "The Information Bottleneck method". The 37th annual Allerton Conference on Communication, Control, and Computing, Sep 1999: pp. 368–377.

[12] M Meila, D Verma, 2001. Comparison of spectral clustering algorithms. University of Washington, technical report.

[13] S.D. Bay, "The UCI KDD Archive," 1999. http://kdd.ics.uci.edu.

[14] CMU. CMU 4 universities WebKB data. 1997.

[15] A.Y. Ng, M.I. Jordan, and Y. Weiss, "On Spectral Clustering: Analysis and an Algorithm," Advances in Neural Information Processing Systems, vol. 14, pp. 849-856, 2001.

AUTHOR'S PROFILE

	**Sondawale Harsha A.**, received the B.E. degrees in Information Technology from Smt. Rajshree Mulak College of Engineering for Women Nagpur, RTMNU in 2012. Now pursuing M.E. in Computer Engineering from K. K. Wagh Institute of Engineering Education & Research, Nashik, India. Her research interests include Machine learning, Image processing, Data Mining.
	**Prof. N. M. Shahane** Associate Professor, Department of Computer Engineering, KKWIEER, Nashik. His research interests include Machine learning, Digital Signal Processing, Digital image processing, Probability & Statistics, Pattern Recognition, Data Mining.